# Micro-Segmentation to reduce threat surface in virtual Data Centre environment against malware proliferation

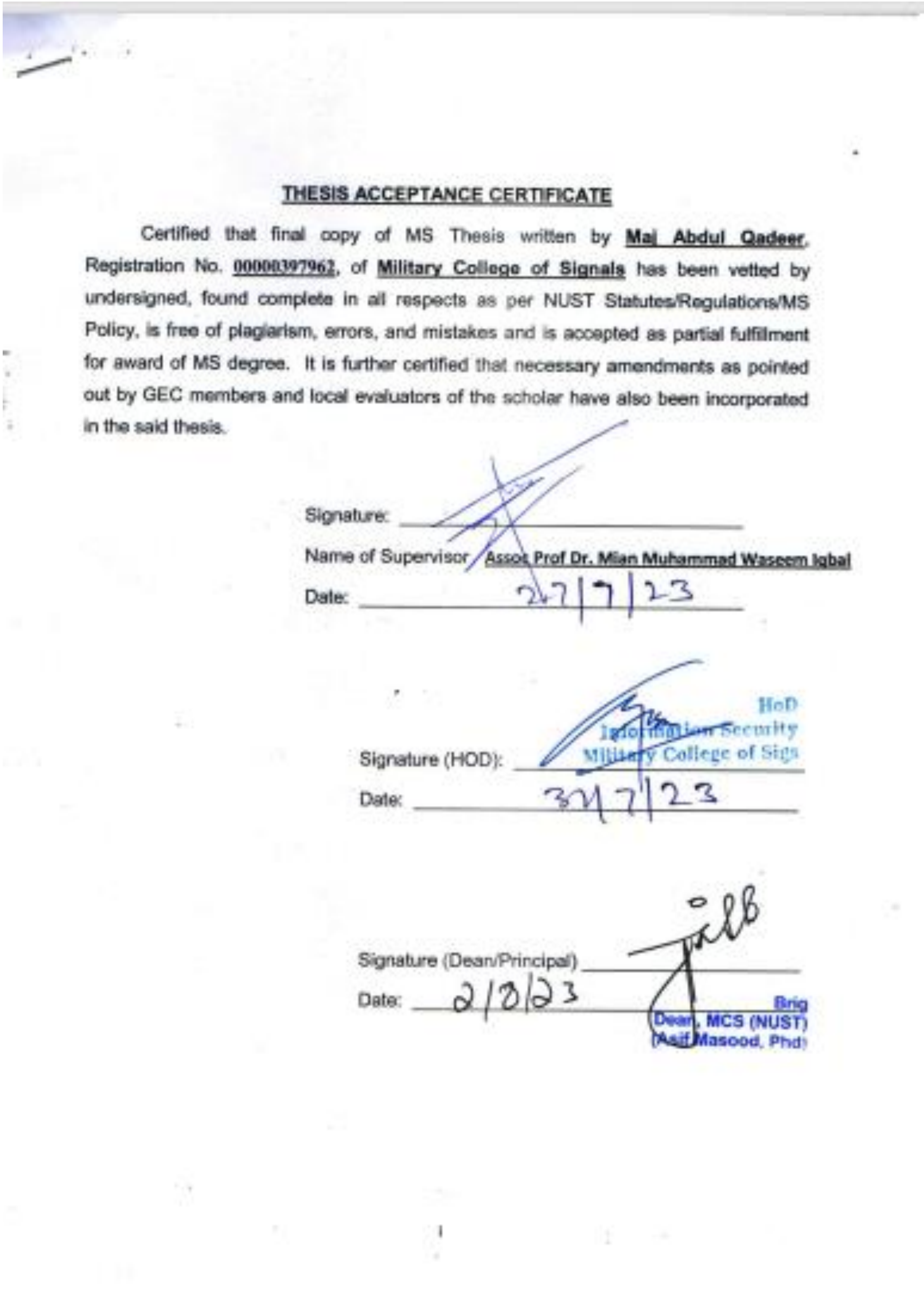
MCS

by

Abdul Qadeer

A thesis submitted to the faculty of Information Security Department Military College of Signals, National University of Sciences and Technology, Rawalpindi in partial fulfillment of the requirements for the degree of MS in Information Security

July 2023



## THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS Thesis written by **Maj Abdul Qadeer**, Registration No. **00000397962**, of **Military College of Signals** has been vetted by undersigned, found complete in all respects as per NUST Statutes/Regulations/MS Policy, is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for award of MS degree. It is further certified that necessary amendments as pointed out by GEC members and local evaluators of the scholar have also been incorporated in the said thesis.

Signature: ______________________

Name of Supervisor **Assoc Prof Dr. Mian Muhammad Waseem Iqbal**

Date: 27/7/23

HoD
Information Security
Military College of Sigs

Signature (HOD): ______________________

Date: 31/7/23

Signature (Dean/Principal) ______________________

Date: 2/8/23

Brig
Dean, MCS (NUST)
(Asif Masood, Phd)

# CERTIFICATE

This is to certify that **Abdul Qadeer** Student of **MS IS** Reg.No **00000397962** has completed his MS Thesis title **"Micro-Segmentation to reduce threat surface in virtual Data Centre environment against malware proliferation "** under my supervision. I have reviewed his final thesis copy and am satisfied with his work.

Thesis Supervisor

(**Assoc Prof Dr. Mian Muhammad Waseem Iqbal)**

Dated: ______Jul 2023

# Declaration

I hereby declare that no portion of work presented in this thesis has been submitted in support of another award or qualification either at this institution or elsewhere

MS Student

(Abdul Qadeer)

Dedicated to:

My Supervisor,

My Committee Members,  
My Family members,  
My Teachers and Colleagues

for their unconditional support, all the way.

# ACKNOWLEDGMENTS

First of all, thanks to Allah Almighty for giving me strength, knowledge and courage tocarry out this research successfully. I am immensely grateful to my supervisor Assoc. Prof. Mian Muhammad Waseem Iqbal for his worthy supervision and support that enabled me to complete my thesiswork. I would also like to thank my committee members, Brig Dr. Imran Rashid, Col Dr. Khawar Mehmood and Col Dr. Imran Makhdoom for their valuable technical support and worthy guidance. Further I am obliged to all my family for their endless support.

# ABSTRACT

The deployment of strong security measures is necessary due to the rise of complex malware lateral threats in fully stacked virtual environment. The effectiveness of micro-segmentation as a tactic to lessen malware's threat surface is examined in this thesis. The study focuses on evaluating the security of virtual machines (VMs) using Windows Defender, the integrated antivirus program in Windows operating systems. Based on this evaluation, dynamic security tags are developed on the software-defined networking platform VMware NSX to categorize virtual machines (VMs) into three separate security groups: infected, protected, and vulnerable. The segmented network's traffic protection measures are then implemented using dynamic criterion policies.

The implementation technique has been described in detail in the thesis, beginning with the gathering of VMs protection status and its current state data. Indicators of malware infections, real-time protection status, OS patched update status, and antivirus signature update status are all included in this data. As a result of the integration of this data with NSX, security tags are automatically issued to VMs, enabling traffic separation and granular security controls.

Through comprehensive testing and analysis of VM behavior within the Vmware Data center environment, the effectiveness of the suggested micro-segmentation approach has been determined. Measured and contrasted against network segmentation methods are metrics like malware containment rates, lateral movement control, and access control enforcement.

Along with the technological implementation, a thorough comparison between network segmentation and micro-segmentation is done as a proof of concept. To fully comprehend the benefits and drawbacks of each strategy, factors including security efficacy, scalability, complexity, performance impact, and auditing capabilities are studied. By using this information, network managers and security experts may choose the best method for strengthening network security in the face of constantly changing malware threats.

Results show how effective micro-segmentation is at reducing malware's attack surface. The detailed comparative analysis and the exact implementation methods add to the body of knowledge previously available on network security. This thesis is a helpful resource for businesses looking to put strong security measures in place and protect their network infrastructures from the malware threats that are continuously changing.

## Table of Contents

Abstract viii

Dedication vi

Acknowledgement vii

List of figures xi

List of tables xii

**Chapter 1**

**Introduction** ………………………………………………………………………. 1

1.1 Overview ……………………………………………………………………… 1

1.2 Motivation………………………………………………………………….. 2

1.3 Challenges in Traditional Network Security Approaches ………………………. 3

1.4 Micro-segmentation……………………………………………………………… 3

1.5 Problem Statement ……………………………………………………………. 4

1.6 Research Objectives ……………………………………………………………... 4

1.7 Contribution ……………………………………………………………………... 5

**Chapter 2**

**Related work and Literature Review** ……………………………………………….. 7

2.1 Micro-Segmentation - Concepts and Components………………………………. 7

2.2 Access Control ………………………………………………………………... 7

2.3 VMware NSX…………………………………………………………………. 8

2.4 Benefits of Micro-Segmentation……………………………………………………. 9

2.5 Implementation Approaches to Micro-Segmentation……………………………. 10

2.6 Hypervisors……………………………………………………………………. 11

2.7 Literature Review………………………………………………………………… 12

**Chapter 3**

**Proposed Methodology**…………………………………………………………….. 17

3.1 Overview ……………………………………………………………………… 17

3.2 Lateral Threat Spectrum ……………………………………………………… 17

3.3 Proposed Strategy……………………………………………………………... 18

3.4 Proposed VM Scanning and VM Tagging Module ……………………………... 22

3.5 VM Scanning Module Algorithm ……………………………………………….. 22

3.6 NSX DFW Dynamic Security Tag Module……………………………………… 24

3.7 Flow Diagram of Proposed Methodology……………………………………... 30

**Chapter 4**

**Results and Analysis** ………………………………………………………………… 32

4.1 Phase 1 ………………………………………………………………………... 32

4.2 Phase 2………………………………………………………………………… 33

4.3 Lateral Traffic Limiting ……………………………………………………… 34

4.4 Data Analysis ………………………………………………………………… 38

**Chapter 5**

**Conclusion** ………………………………………………………………………….. 44

5.1 Objectives Achieved………………………………………………………… 44

5.2 Limitations………………………………………………………………….. 45

5.3 Future Direction…………………………………………………………….. 45

# List of Figures

3.3 Proposed Architecture 20

3.3.1 VCSA Environment Summary 21

3.3.2 Hosts and Guest VMs 21

3.3.1 VMs Security State Transition 23

3.3.2 NSX Dynamic Security Tag Assignment 24

3.5.5 POC Lab Environment 27

3.5.6 NSX Manager Summary 28

3.5.7 NSX to VCSA Integration 29

3.5.8 NSX & Hosts 30

3.7 Flow Chart Proposed Methodology 30

4.1.1 Machine Status 31

4.1.2 List VMs Scanned successfully 32

4.2.1 VMs Security Tags Allocation 33

4.2.2 VMs Assigned to Associated Security Tag 33

4.2.3 Security Groups Dynamic Criteria based Assignment of Security Tags 33

4.2.4 Service Composer Dynamic Collection of VM into a Single Container 34

4.2.5 DFW Rules applied to Security Groups as per Matrix 35

4.3.1 ICMP Status Protected_VM to Protected_VM 35

4.3.2 ICMP Status Protected_VM to Vulnerable_VM 35

4.3.2a ICMP Wireshark Status Protected_VM to Vulnerable_VM 36

4.3.3 ICMP Status Vulnerable_VM to Protected_VM 36

4.3.3a ICMP Nil Wireshark Status Vulnerable_VM to Protected_VM 37

4.3.4 Isolated VMs Policy Status (Infected_VMs) 38

5.3.1 Malware Contained in Network Segment Vs Micro-Segment 38

5.3.2 Lateral Propagation Analysis Chart 39

5.3.2 Micro-Segmentation Vs Network Segmentation 42

# List of Tables

2.0 Survey Report 14

3.6.1 Service Composer 25

3.6.2 Security Groups Services Matrix 26

4.1.2.1 CSV VMs Protection Status 32

4.4.3.1 Malware Contained 37

4.4.3.2 Malware Propagation 38

4.4.5 Implementation of Controls 40

4.4.6 Collection of Incident Data 41

4.4.7 Sliding Scale Micro-Segmentation vs Network Segmentation 41

*Chapter 1*

# INTRODUCTION

## 1.1. **Overview**

The increasing and expanding modern assaults on data Center information assets is a cause for concern. Attackers have discovered inventive ways to penetrate the data center. It takes considerable time to exploit north-south perimeter defense and later utilize inside communication pathways to reach their craved targets and compromise them by quick lateral movement [1][3]. The conventional security approach depended fundamentally on border defense by securing the north-south activity, but expecting that east-west activity within the information center is normally secure. In legacy architecture the security through perimeter firewall achieved by configuring security zones and segmenting the layer-3 (IP Segments) and layer-2 (VLANs) domains [3]. It becomes more challenging once coming into virtual compute environment where multiple virtually separated segments are operating within same host; hence very little control left with traditional firewall security flows. Especially when pool of resources allocated, deallocated or transferred within or across infrastructure security management is more challenging. Micro-Segmentation is one approach to improve virtual data center defenses. This approach can do inspection of every traffic flow within the data center at granular level. Micro-Segmentation can control the data center infrastructure up to vNIC (Virtual Machine interface) level. Micro-segmentation is very useful to stop lateral movement of malware and reduce the threat surface once passed through perimeter FW security.

Compute virtualization is now increasingly functioning in data center environment and over the period of time it's getting mature and definitely helped to increase efficiency and enhance scalability. Network models have also evolved to match the efficient computer resource. As in virtualize environment network has to work at hypervisor level and no spine and leaf model [4]. The castle defense has failed to deliver the promised boundary security as once boundary perimeter crossed no control on lateral threat. Each time an unused innovation is put in to counter found misuses, it makes the front entryway more complicated, costlier, and more defenseless. We must expect that dangers may be show within the framework at any time, and indeed with discovery and moderation, we must expect proceeded risk nearness over long periods. Malware is always an alive threat to data center compute; however, there are virtual environment is different from traditional data center servers as malware have to detect the virtual environment first to execute differently.

Malware analysis techniques in virtual compute environment are also different as compared to traditional malware analysis; as malwares now a day first distinguish the environment in which they are running. Cyber-kill chain is a concept which both attacker and defender follows for respective portfolio, it has seven steps that followed by attacker to execute the attack and as defender we have to break the chain [4]. The important task in attacker's perspective is delivery and exploitation of payload that carries malware. If attacker succeeded in exploitation, it has now access to the targeted system and then he will raise privileges by next phase that is installation. After successfully executing the installation phase attacker wish to propagate into other systems in same virtual environment e.g. east-west propagation and at this point of time no traditional firewall can stop the further damage. The legacy approach has no reply for this worldview, so we must utilize a distinctive approach. In this paper we will address the lateral threat mitigation and implement the concept in VMware NSX micro-segmentation.

## 1.2. **Motivation**

### 1.2.1. **The Need for Advanced Network Security**

In today's interconnected and digitized world, organizations face increasingly sophisticated and persistent cyber threats. Traditional network security approaches, such as perimeter-based defenses, are no longer sufficient to protect against these evolving threats. There is a growing need for advanced network security techniques that provide granular control over network traffic and minimize the potential impact of security breaches.

The need for advanced network security arises from several factors. First, the threat landscape is constantly evolving, with cybercriminals employing advanced techniques to bypass traditional security measures. Organizations must adapt to these evolving threats and adopt proactive security measures that can effectively detect and mitigate attacks.

Second, the reliance on perimeter-based security models, such as firewalls and intrusion detection systems, is no longer adequate. These approaches assume that the network perimeter can be effectively defended, but with the proliferation of cloud computing, mobile devices, and remote work, the network perimeter has become porous and less well-defined. This creates new entry points for attackers and increases the attack surface.

Finally, compliance and regulatory requirements impose strict security standards on organizations, particularly in industries handling sensitive data such as healthcare and finance. Advanced network security measures are necessary to ensure compliance with these standards and protect sensitive information from unauthorized access.

## 1.3. Challenges in Traditional Network Security Approaches

Traditional network security approaches face several challenges in effectively addressing the evolving threat landscape. These challenges include [5][6]:

1.3.1. **Lateral Movement** Ineffective safeguards to stop lateral movement within the network are frequently absent from traditional security approaches. After gaining access to the network, an attacker is free to move around and investigate the internal systems, escalating privileges and accessing confidential information.

1.3.2. **Limited Visibility** Users and network activity are not always visible when using traditional security techniques. This lack of visibility makes it difficult to identify and address security incidents in a timely manner.

1.3.3. **Reliance on perimeter** The assumption made when relying entirely on perimeter-based defenses is that all attacks can be stopped at the network edge. However, enterprises require more precise security measures to safeguard crucial assets due to the rise in sophisticated assaults and insider threats.

1.3.4. **Complexity and Maintenance** Traditional security architectures often involve complex configurations, multiple security devices, and manual rule management. This complexity introduces administrative overhead and makes it challenging to maintain an effective security posture.

1.3.5. **Introduction to Micro-Segmentation** The limitations of conventional security methods are addressed by the sophisticated network security technology known as micro-segmentation. It entails segmenting a network into more manageable, isolated chunks, allowing businesses to impose stringent access controls and carry out security procedures at a more granular level.

1.4.**Micro-segmentation** It goes beyond perimeter defenses and focuses on securing individual network segments or even specific devices or applications. By implementing micro-segmentation, organizations can minimize the lateral movement of threats, reduce the attack surface, and enhance overall network security. Creating security zones within the network, where access between segments is regulated based on predetermined regulations, is the fundamental idea of micro-segmentation. Using technologies like virtualization and Software-Defined Networking (SDN), segmentation can be accomplished, enabling dynamic and adaptable security measures [6]. Numerous advantages of micro-segmentation include increased network security, a smaller attack surface, regulatory compliance, improved incident response, scalability, and performance optimization. Micro-segmentation implementation, meanwhile, is not without its difficulties. These include implementation complexity, network visibility, and infrastructure integration. To successfully implement micro-segmentation as a good network security solution, several issues must be properly addressed.

Lateral threat surface mitigation is always a challenge and it becomes more complicated in virtual data center environment. This effort is being carried out to utilize the advantages of Micro-Segmentation in distributed firewall for virtual computing and networking environment to reduce lateral threat surface for malware proliferation. Lateral threat surface is always cause of concern for securing data center internal servers communication [3][7].

Traditional defense in depth and castle security architecture cannot secure east-west flow of traffic. Compute resource virtualization further complicate the security situation where traditional network segmentations are no more effective specifically in view of network virtualization. There is need of a comprehensive framework to protect lateral surface in virtual environment under the umbrella of zero-trust which is specifically important for virtual data center environment. Micro-segmentation is one of the mainstay of zero-trust model.

## 1.5. **Problem Statement**

The castle architecture dually protected with D-n-D is failing against modern day security challenges. Traditional in-house solutions are inadequate to ensure the security specially in fully virtualized environment. Securing the lateral East-West movement is always a challenge with traditional Firewalling that can only protect North-South traffic through network segmentation. Hence, always there is a need to develop a security mechanism to protect lateral East-West movement within virtualized environment.

## 1.6. **Research Objectives**

The main objectives of this thesis are:

- To study the current models and techniques available to protect lateral threat surface of virtual data center environment. Specifically, under virtual network how these threats are protected.
- To propose a centralized framework that can protect east-west lateral threat surface under SDN (NSX) architecture using micro-segmentation to create Distributed policies to protect malware prorogation.
- Comparative competitive analysis of the proposed technique with the existing solutions available.

## 1.7. **Contribution**

- Provide more effective and efficient security for inter application communication in virtual computing.
- Minimize the occurrence of security breaches to a negligible level.
- Increase stakeholders' confidence in the system.
- Decrease dependencies on traditional castle defense.

## 1.8. **Thesis Outline**

The research work has been organized and distributed in following chapters:

- **Chapter 1:** A brief introduction is given, motivation and research objectives are highlighted. Problem statement is given furthermore contribution made through this research is enumerated.
- **Chapter 2:** This chapter covers the core concepts related to Micro-segmentation in specific and in general covering related aspects. Second part of this chapter covers literature review.
- **Chapter 3:** This chapter covered the proposed methodology in detail. It has proposed architecture block diagram and its explanation, then It covers the proposed solution algorithms and its flow diagram. POC environment where proposed concept has been verified discussed in detail.

- **Chapter 4:** In this section a comprehensive analysis has been carried out along with results from POC environment has also been shown both statistically and graphically.
- **Chapter 5:** It has conclusion of research along with objectives achieved. It has also covered shortcomings and benefits. Finally, future work discussed.

*Chapter 2*

# Related work and Literature Review

## 2.1 Micro-Segmentation - Concepts and Components

In order to establish finer control over network traffic, a network is divided into smaller, isolated segments using the sophisticated network security technique known as micro-segmentation. Segmentation, access control, and policy enforcement are its guiding principles. Segmentation: Micro-segmentation breaks down the network into smaller security zones or segments, where each segment represents a distinct area with specific security requirements [8] [11] [12]. This segmentation prevents lateral movement of threats and contains potential security breaches within isolated segments. The salient of this technology are: -

2.2 **Access Control** Within each network segment, rigorous access constraints are imposed by micro segmentation. Organizations can specify which individuals, gadgets, or apps are permitted to communicate both within and between segments by using it to build fine-grained access restrictions [9]. By restricting access to vital resources, this strategy reduces the attack surface. Policy Enforcement: Micro-segmentation relies on the enforcement of security policies at the segment level. These policies define the desired behavior and restrictions for communication between segments. Policy enforcement mechanisms ensure that network traffic adheres to the defined security policies and any violations are promptly detected and addressed [12].

2.2.1 **Micro-Segmentation Architecture** The design and organization of the network infrastructure required to implement micro-segmentation are included in micro-segmentation architecture. Although the specific architecture may change depending on the needs of the business and its current infrastructure, the following core components are always present:

2.2.1.1 **Control Plane** The configuration and communication between network devices are handled by the control plane. It manages traffic flow, upholds security regulations, and makes sure access controls are uniformly enforced throughout all network segments.

2.2.1.2 **Management Plane** Network administrators can configure and keep track of the micro-segmented network using a centralized interface provided by the management plane. It enables the design of policies, traffic flow monitoring, and application of security measures.

2.2.1.3 **Components of Micro-Segmentation** Several components are essential to the architecture of micro-segmentation in order to implement it successfully. Together, these elements set security rules, enforce access controls, and maintain the appropriate level of network security.

2.2.1.4 **Software-Defined Networking (SDN)** SDN is a technology that divides the control plane and data plane of a network, allowing for network centralization and programmability. By enabling dynamic control over network traffic, policy enforcement, and traffic routing, SDN offers the agility and flexibility required to achieve micro-segmentation.

2.3 **VMware NSX** A network virtualization and security technology called VMware NSX enables businesses to build virtual networks and apply micro-segmentation to their infrastructure. By separating network services from the underlying physical hardware, NSX makes it possible to build virtual networks that can be separately controlled and secured. Main components used in this research are following: -

2.3.1 **Distributed Firewall** Distributed Firewall (DFW) of NSX is a crucial part that offers the ability to micro-segment networks. It works at the hypervisor level, allowing for fine-grained monitoring and control of traffic between virtual machines (VMs) at the network's edge. Organizations can specify granular security controls and regulate communication both within and across segments thanks to the DFW's application of firewall rules at the VM level.

2.3.2 **Service Composer** The service composer from NSX is a tool for managing policies that makes setting up and enforcing security policies simple. For setting up security groups, firewall rules, and other security services, it offers a consolidated interface. Service Composer enables consistent and effective policy management by allowing administrators to develop reusable security policies and apply them to particular VMs or groups of VMs.

2.3.3 **Security Groups** The idea of security groups, which are logical groupings of VMs based on shared features, is used by NSX. Various characteristics, including IP address, operating system type, or application kind, can be used to form security groups. Administrators can design and implement security policies at the group level by associating VMs with particular security groups, which makes it easier to manage and scale micro-segmentation rules.

2.3.4 **Service Insertion** Through service insertion, NSX enables the incorporation of external security services into the micro-segmented environment. This makes it possible for businesses to use extra security features provided by outside suppliers, such as intrusion detection and prevention systems (IDPS), advanced threat protection, or data loss prevention (DLP) solutions. Service insertion makes sure that these security services may check traffic moving across micro-segments, improving the overall security posture.

## 2.4 Benefits of Micro-Segmentation

Numerous advantages of micro-segmentation include increased network security, decreased attack surface, assurance of regulatory compliance, improved incident response capabilities, and optimized scalability and speed. Because of these advantages, micro-segmentation is becoming a more common strategy for businesses trying to improve their network security posture.

2.4.1 **Enhanced Network Security** Network security can be improved through micro-segmentation, which is one of its main advantages. Micro-segmentation restricts the lateral flow of threats within the network by breaking the network up into more manageable, isolated sections. The potential impact of a security breach is diminished if an attacker gains access to one segment since their capacity to move laterally to other segments is severely constrained [8]. By adding another line of defense, this isolation makes it harder for attackers to gain access to sensitive data or elevate their privileges.

2.4.2 **Reduced Attack Surface** The attack surface of an organization's network is dramatically decreased via micro-segmentation. The number of potential targets for attackers is reduced by dividing the network into more manageable, separate regions. Even if one section is infected, the assault is limited to that segment and doesn't spread to other parts of the network [15]. By lowering the attack surface, organizations may reduce the potential damage from security breaches, slow the propagation of malware, and keep tabs on their most important assets.

2.4.3 **Compliance and Regulatory Requirements** Organizations can achieve and maintain regulatory compliance and industry-specific criteria with the help of micro-segmentation. Strict access restrictions and data protection procedures are required by many regulations, including the General Data Protection Regulation (GDPR) and the Payment Card Industry Data Security Standard (PCI DSS) [20]. By isolating sensitive data within designated segments and implementing particular security policies to adhere to regulatory standards, micro-segmentation enables enterprises to adopt and execute these security controls. Organizations can avoid fines, safeguard their brand, and increase customer trust by proving compliance.

2.4.4 **Improved Incident Response** By isolating security issues within discrete segments, micro-segmentation improves incident response capabilities. Organizations can immediately recognize and address threats within the impacted segment after an incident, minimizing attacker lateral movement and reducing the potential impact on the rest of the network. With less overall reaction time and less disruption from security breaches, this isolation enables security teams to analyze and mitigate incidents more effectively. Additionally, enterprises may better monitor and detect unusual activity, enabling proactive threat identification and response, with the help of granular access controls and visibility into network traffic inside each segment.

2.4.5 **Scalability and Performance Optimization** Scalability and performance improvement are advantages of micro-segmentation. Organizations can grow their security measures more successfully by segmenting the network. Security policies and controls can be created and managed at the segment level, providing for more flexible and effective security management as opposed to securing the entire network as a single, monolithic entity. This scalability makes sure that security policies can be modified and applied to new segments as the network expands or changes without affecting the infrastructure of the entire network.

## 2.5 Implementation Approaches to Micro-Segmentation

Micro-segmentation reduces the attack surface to a minimum, introduces access controls to isolated segments, and allows organizations to monitor and control traffic to each segment. Micro-segmentation has three main approaches, which vary depending on which network layer is used for implementation.

2.5.1 **Network-Based Micro-Segmentation** Switches, routers, and load balancers are just a few examples of the network-based micro-segmentation tools that are used as enforcement points. The network is divided into sections using VLANs, subnets, firewalls, or some other kind of tagging technology. The majority of network traffic that uses this strategy is North–South traffic. Micro-segmentation solutions are limited by the capabilities of the network device of the manufacturer; as a result, they cannot be used in data centers or cloud platforms that employ devices from different network equipment vendors [13] [18]. This strategy has several drawbacks, including the creation of macro-segmentation, the division of a network into numerous distinct portions to serve business needs; instead of micro-segmentation and the high cost of implementation.

2.5.2 **Hypervisor-based Micro-Segmentation** A piece of hardware, software, or firmware known as a hypervisor may build virtual computers, manage them, and allocate resources to them. Virtual workloads and software-defined data centers are increasingly becoming the standard in contemporary data centers. Since the hypervisor must be used for all workload traffic, network isolation and micro-segmentation can be done there. This method makes use of the firewall's ability to provide visibility and micro-segment workloads.

2.5.3 **Host-based Micro-Segmentation** When hosts are deployed, micro-segmentation rules may be added, and they will thereafter stay with the workloads regardless of their location or duration. This is especially useful for workloads that are temporal and mobile. With host-based technologies, it might be challenging to manage policy and enforcement rules for hundreds or thousands of workloads rather than a few centralized networks or hypervisors.

## 2.6 **Hypervisors**

Multiple operating systems can run independent applications on the same server while sharing the same physical resources thanks to server virtualization. System and network administrators can now have a dedicated computer for each service that needs to be run thanks to virtual machines. A hypervisor is placed on a host machine, as opposed to the guest virtual machines that run on top of it. The hypervisor is a kind of hardware virtualization that enables many guest operating systems to run simultaneously on a single host computer.

### 2.6.1.1 **Types of Hypervisor**

- **Type-1 Hypervisor** A Type 1 hypervisor that works directly on the host computer's physical hardware is known as bare-metal or native. The loading of an operating system is not necessary for the Type 1 hypervisor. Direct access to the underlying hardware and no additional software to think about make virtualization simpler. Hypervisors that run directly on actual hardware are also quite safe. Virtualization lessens the risk of attacks that target OS security flaws and vulnerabilities because each guest has its own OS [16].

- **Type-2 Hypervisor** A Type 2 hypervisor is often installed on top of an active operating system. It's also known as a hosted hypervisor since it uses the host machine's pre-existing OS to control calls to CPU, memory, storage, and network resources. Because of this, Type 2 hypervisors are frequently not used in data centers; rather, they are saved for client or end-user systems (also known as client hypervisors), where performance and security are less of an issue. When compared to production virtualized systems or the cloud, they are most frequently used in configurations with a small number of servers and are less expensive than Type 1 hypervisors, making them a good test platform.

## 2.7 Literature Review

A number of researchers have carried out their research on malware analysis to propose mitigation techniques. Zero-trust protection model has also remained under focus of many researchers in recent years, however micro-segmentation has been touched by few. Mitigation solutions based on static malware analysis and based are more as compare to dynamic analysis. Most of the work carried out with dynamic analysis technique covered to identify the Behavior of specific malware rather worked on propagation. In this chapter I have covered most of the work related to zero-trust based mitigation against malware with mainly focus on Micro-segmentation.

## 2.7 Micro-segmentation Protection Scheme Based on Zero Trust Architecture

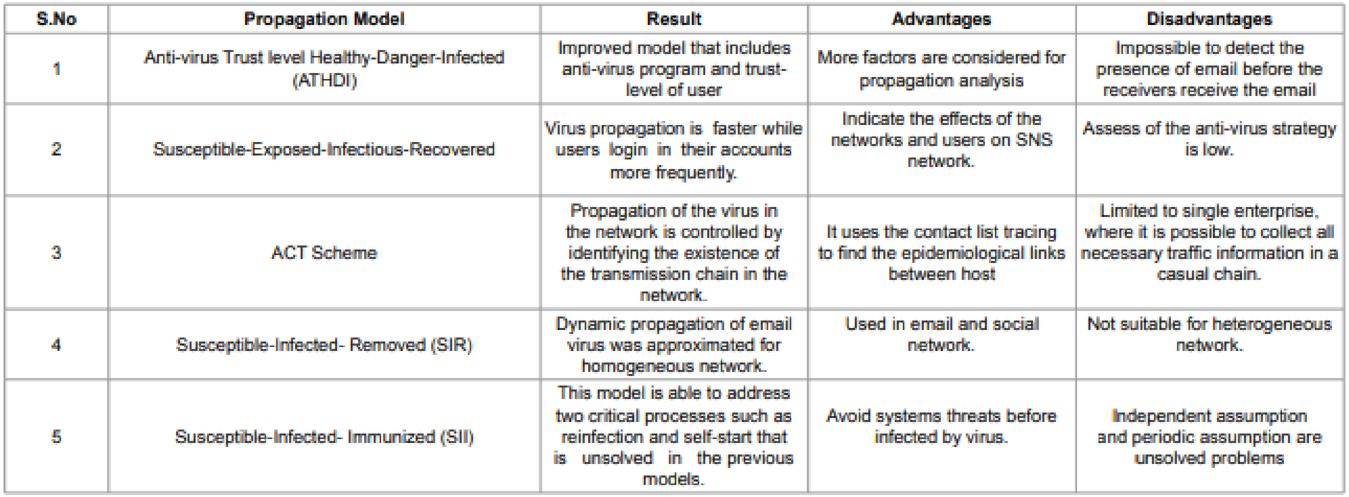
It offered a Micro-Segmentation-based network protection mechanism, which focused on identifying abnormal behavior in network traffic and using that information to identify abnormal behavior devices connected to that network in order to protect east-west traffic using adaptive network traffic control. Once the abnormal device was detected, the protection module immediately limited access privileges. The security gateway was used to protect north-south traffic [3] [11].

## 2.8 Automated Micro-segmentation for Lateral Movement Prevention in Industrial Internet of Things (1IoT)

In this research idea is to protect attacks within IOTs devices, they primarily focused on industrial IOTs. They used SCADA to collect data from central point and then use machine learning techniques to analyze traffic to make real-time decisions on communication of IIOTs. In this research the threat model in based on lateral movement of attacker within IIoTs [21].

## 2.9 A Survey on Malware Propagation Analysis and Prevention Model

I am putting this survey here with purpose to get fair idea about malware propagation and proposed mitigation techniques. Different methodologies have been adopted by different researchers, I am sharing their results here to show a summarized result. The main focus of all these techniques to identify malware propagation due to emails [22].



| S.No | Propagation Model | Result | Advantages | Disadvantages |
|---|---|---|---|---|
| 1 | Anti-virus Trust level Healthy-Danger-Infected (ATHDI) | Improved model that includes anti-virus program and trust-level of user | More factors are considered for propagation analysis | Impossible to detect the presence of email before the receivers receive the email |
| 2 | Susceptible-Exposed-Infectious-Recovered | Virus propagation is faster while users login in their accounts more frequently. | Indicate the effects of the networks and users on SNS network. | Assess of the anti-virus strategy is low. |
| 3 | ACT Scheme | Propagation of the virus in the network is controlled by identifying the existence of the transmission chain in the network. | It uses the contact list tracing to find the epidemiological links between host | Limited to single enterprise, where it is possible to collect all necessary traffic information in a casual chain. |
| 4 | Susceptible-Infected- Removed (SIR) | Dynamic propagation of email virus was approximated for homogeneous network. | Used in email and social network. | Not suitable for heterogeneous network. |
| 5 | Susceptible-Infected- Immunized (SII) | This model is able to address two critical processes such as reinfection and self-start that is unsolved in the previous models. | Avoid systems threats before infected by virus. | Independent assumption and periodic assumption are unsolved problems |

**Table 2.0 Survey Report**

## 2.10 A Method for Malware Detection in Virtualization Environment

Future computing models may include cloud computing. A crucial foundational technology for cloud computing is virtualization. Using Virtual Machine Monitor (VMM) or a hypervisor, virtualization builds and executes numerous virtual machines (VM) or guest operating systems on a single physical machine. Hypervisor makes abstracting from physical machines easier. Many virtual machines can share resources like CPU, Memory, I/O, and NIC, among others [23]. The cloud service provider faces more security issues as a result of resource sharing. More frequently, sophisticated rootkits and the spread of unidentified malware aim to alter crucial kernel data structures. Particularly in a virtualized environment, a traditional in-host anti-malware solution is insufficient to guarantee the security of the guest operating system. In this paper, researchers have proposed a malware detection technique based on analysis of API function calls and API function call sequences. Technique used track calls to API functions and function call patterns indicating different sorts of attacks using process injection [23]. To represent the retrieved API function calls as a feature in the machine learning model. On Windows virtual machines, various malware injectors are run and their runtime memory is gathered. A volatility framework is used to conduct behavior-based dynamic analysis.

## 2.11 Dynamical Propagation Model of Malware for Cloud Computing Security

This is very pertinent research that actually covered the aspect of malware propagation in a virtualized environment. This research proposed a mathematical model that covers the malware propagation probability in virtual data center based on different states of VMs. I have used same model as base of my proposed malware mitigation technique as; I have incorporated micro-segmentation with this model [9].

## 2.12 Fibonacci Modeling of Malware Propagation

In this research they have use Fibonacci rabbit problem as base model to make propagation sequence and the mathematical model is based on Fibonacci number sequence. They have modeled the various malware scanning techniques, how they get vulnerable hosts [24]. They have manly focused on propagation time of malware in between the hosts and performed experiments to evaluate the effects of diverse propagation times on the malware propagation.

## 2.13 Investigating Malware Propagation and Behaviour Using System and Network Pixel-Based Visualization

This research first examines the design elements of presenting this data in a scalable manner for a security analyst, after which we detail the experimental setup for our data gathering. Malware analysis frequently makes use of sandbox environments, which are frequently achieved utilizing a virtual machine to investigate the behaviors of malware when executed [25]. Cuckoo Sandbox is a great tool for malware investigation, but Cuckoo's focus on a single virtual environment rather than tracking malware spread throughout workstations limits its use. So they focus to create a unique environment based on Cuckoo that supports and collects data from various targets and analyze behavior centrally.

## 2.14 Micro-segmentation: securing complex cloud environments

The datacenter as we know it has undergone such a profound upheaval in recent years. This paper briefly covers security challenges and emphasis on zero-trust model. Micro-segmentation is one of the pillar to achieve the zero-trust in virtual data centers [12]. This paper helps me to understand what are pre-requisites of micro-segmentation, how it works and what are the possible barriers to implement in data center environment. It also covers the best practices and use cases of real world data centers.

## 2.15 SDN-Based Detection of Self-Propagating Ransomware: The Case of BadRabbit

The most significant and current SDN-based studies on malware detection in general and ransomware detection in particular are reviewed in this paper. Researchers were motivated by the flexibility that SDN delivers and the associated controller programmability. Their proposed solution is based on detecting mobile malware by assuming that mobiles are connected to access point SDN switches [4]. They have focused on detection of ransomware. The goal of this research was to prevent BadRabbit from spreading itself. To identify and stop self-replicating malware like BadRabbit, they have created an SDN-based IDPS.

## 2.16 THE V-NETWORK: A TESTBED FOR MALWARE ANALYSIS

In this research, a virtualized environment known as the V-Network was described. It was used to test countermeasure systems and analyze network worm spread. In the V-Network testbed, there are 1200 virtual machines. A number of virtual networks. An old-fashioned worm outbreak scenario and a modern worm scenario based on the 2014 Shell-Shock vulnerability were used to test the effectiveness of V-Network [26]. The findings demonstrate that the V-Network testbed is a reliable and practical platform for studying malware propagation. The main focus was to propose a reliable environment for malware analyzing.

*Chapter 3*

# PROPOSED METHODOLOGY

## 3.1 **Overview**

Adversaries utilize lateral movement as a method to increase their network access in order to move through an environment and accomplish their objectives. These objectives include gaining access to servers hosting sensitive data or applications, running malicious code on them, and then enlarging the attack surface to the east and west.

Targeting on-premises networks based on network protocols and services like Active Directory and NTLM has been done for years using lateral movement. Lateral movement has been used in many significant attacks e.g. Stuxnet worm that spread over network [17]. When an attacker successfully compromises a workload in a cloud environment, they will go for lateral movement tactics to misuse IAM rights (privilege escalation) or jump from one workload to another within virtual environment.

## 3.2 **Lateral Threat Spectrum**

New security issues, such as the risk of malware lateral movement, have surfaced with the growing usage of virtualization technologies and the proliferation of virtual data centers. Once malware has obtained entry to a virtual data center, lateral movement describes how it might spread horizontally across systems and networks [27]. The security and integrity of the virtualized environment are seriously jeopardized by this lateral migration.

### 3.2.1 **Exploiting Shared Resources**

Common shared resources used by virtual data centers include networking, storage, and hypervisors. Malware that infects a single virtual machine (VM) within the data center can use these shared resources to move laterally across other linked VMs [27]. Malware can travel covertly from one virtual machine to another by taking advantage of flaws in the virtualization layer or incorrect setups, circumventing standard security procedures.

### 3.2.2 **Weak Segmentation and Access Controls**

Within a virtual data center, poor segmentation and access controls may facilitate malware's lateral movement. Malware can penetrate VMs, subnets, or security zones if network and security policies are not properly specified and implemented, thereby exposing sensitive data and crucial systems. Fine-grained or granular segmentation and access controls must be established in a virtual environment to prevent lateral movement due to the dynamic nature of VMs. Mare network segments cannot limit the threat surface after exploitation and installation of malicious code inside VM [27].

3.2.3 **Network Traffic Visibility** It might be difficult to identify and monitor the lateral movement of malware in virtualized systems because of the frequently complicated network topologies and high traffic volumes that they feature. The capacity to identify and counteract lateral movement operations may be constrained by the limited visibility that traditional network security tools and monitoring systems are able to provide into inter-VM communication. Malware's potential to wreak more harm by going undetected may be due to this lack of visibility.

3.2.4 **Privilege Escalation** Malware can attempt to escalate privileges and get administrative access or elevated privileges within the virtual data center if it is able to access a single VM. The malware can travel laterally to other VMs with higher privileges through privilege escalation, possibly compromising crucial infrastructure parts and sensitive data repositories [27]. To reduce the danger of privilege escalation, proper privilege management and access control measures are crucial.

3.2.5 **Remote Service Session Hijacking** Adversaries may take over active sessions with remote services in order to move laterally within a system. Users can use valid credentials to log into a service designed specifically to accept remote connections, such as telnet, SSH, and RDP [27]. When a user connects into a service, a session is created that enables ongoing interaction with it.

3.2.6 **SSH Hijacking** Adversaries may utilize a genuine user's SSH session as a jumping off point to travel laterally through a system. Secure Shell (SSH) is the industry-standard method of remote access. Through the use of an encrypted tunnel and typically through the use of a password, certificate, or asymmetric encryption key pair, it enables users to connect to other systems [27].

3.2.7 **Replication Through Removable Media** Adversaries may enter systems, possibly those on disconnected or air-gapped networks, by copying malware onto portable media and using autorun features when the media is inserted into a system and runs [27]. In the case of lateral movement, this may take the form of malware that has been copied and renamed to trick users into running it on a different machine or the modification of executable files stored on portable media.

## 3.3 **Proposed Strategy**

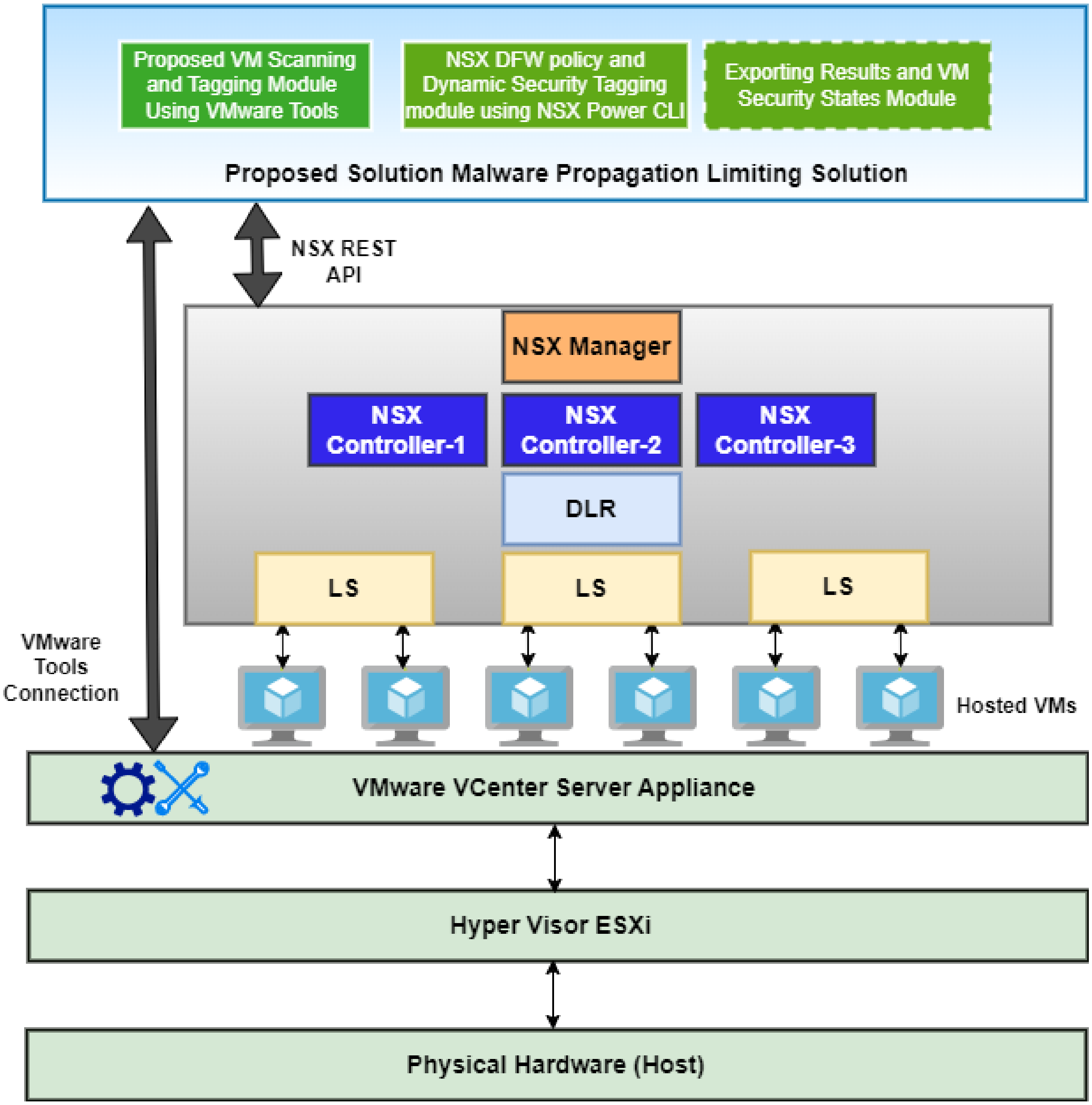
My proposed model derived from the term Zero Trust where we do not trust any of the entity starting from IAM to hardening of the VM and physical controls of the environment. Micro-Segmentation is one of the pillar of zero trust model. I have used micro-segmentation to reduce above mentioned threat spectrum to escalate latterly within virtual environment in any enterprise data center. Proposed solution is specific to limit malware lateral propagation.

### 3.3.1 Proposed Framework Architecture

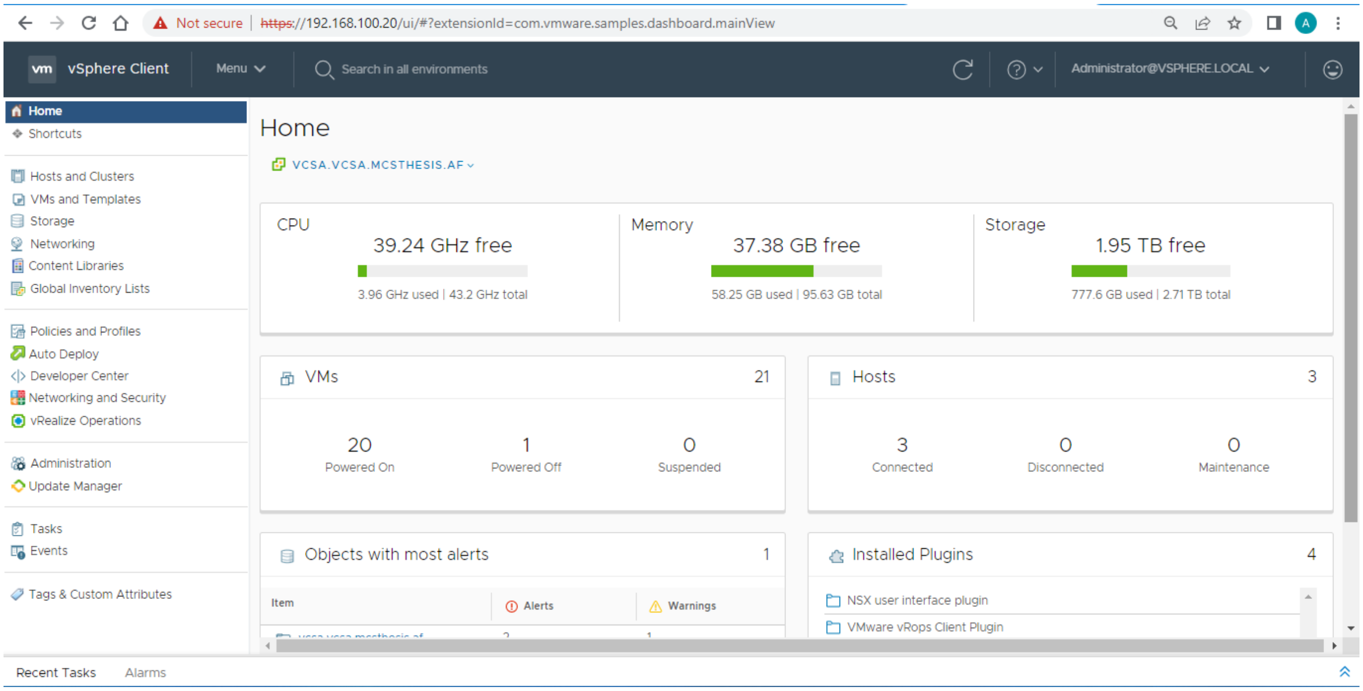
The proposed architecture, as shown in Figure 4.3, is designed for a completely virtualized environment deployed within a virtual data center. By combining compute and network virtualization technologies, this architecture enables effective management and improved security through micro-segmentation. knowledge the full environment stack is crucial to provide a comprehensive knowledge of the proposed architecture. The three ESXi Hosts that make up the deployed environment stack all function as Type-1 Hypervisors. Multiple virtual machines (VMs) have been created and managed by these hypervisors within the virtual data center. The installation of the VMware vCenter Server Appliance allows for the centralized management of the virtualized compute resources. The ESXi Hosts, VMs, and other related resources may all be managed from a single platform VMware VCSA. For provisioning, monitoring, and configuring the virtual infrastructure, it provides complete features.

A number of Windows 10 VMs have been deployed and dispersed among the three ESXi Hosts in order to provide the necessary data sets for the proposed model. The many elements and workloads (AD, Web, App, DB) that make up the virtual data center environment are represented by these VMs.

The architecture includes the installation of NSX, VMware's network virtualization and security platform, over the top of the virtualized environment. In order to ensure strong network security inside the virtual data centre, NSX enables the development of secure connectivity and micro-segmentation among the VMs. All of the deployed VMs are connected through NSX. By isolating network segments and imposing granular access restrictions and security policies, NSX's dynamic micro-segmentation capabilities boost security. Hence; Micro-segmentation's goal of providing security is in line with the proposed design. The architecture effectively restricts lateral movement and contains any security breaches inside particular parts by segmenting the virtualized network into smaller, isolated pieces. The attack surface is greatly reduced by this isolation, which also improves the virtual data center environment's overall security posture.


Proposed VM Scanning and Tagging Module Using VMware Tools
NSX DFW policy and Dynamic Security Tagging module using NSX Power CLI
Exporting Results and VM Security States Module
Proposed Solution Malware Propagation Limiting Solution
NSX REST API
NSX Manager
NSX Controller-1
NSX Controller-2
NSX Controller-3
DLR
LS
LS
LS
VMware Tools Connection
Hosted VMs
VMware VCenter Server Appliance
Hyper Visor ESXi
Physical Hardware (Host)

**(Figure: 3.3 Proposed Architecture)**



Home

VCSA.VCSA.MCSTHESIS.AF

| CPU | Memory | Storage |
|---|---|---|
| 39.24 GHz free | 37.38 GB free | 1.95 TB free |
| 3.96 GHz used \| 43.2 GHz total | 58.25 GB used \| 95.63 GB total | 777.6 GB used \| 2.71 TB total |

VMs 21

| 20 | 1 | 0 |
|---|---|---|
| Powered On | Powered Off | Suspended |

Hosts 3

| 3 | 0 | 0 |
|---|---|---|
| Connected | Disconnected | Maintenance |

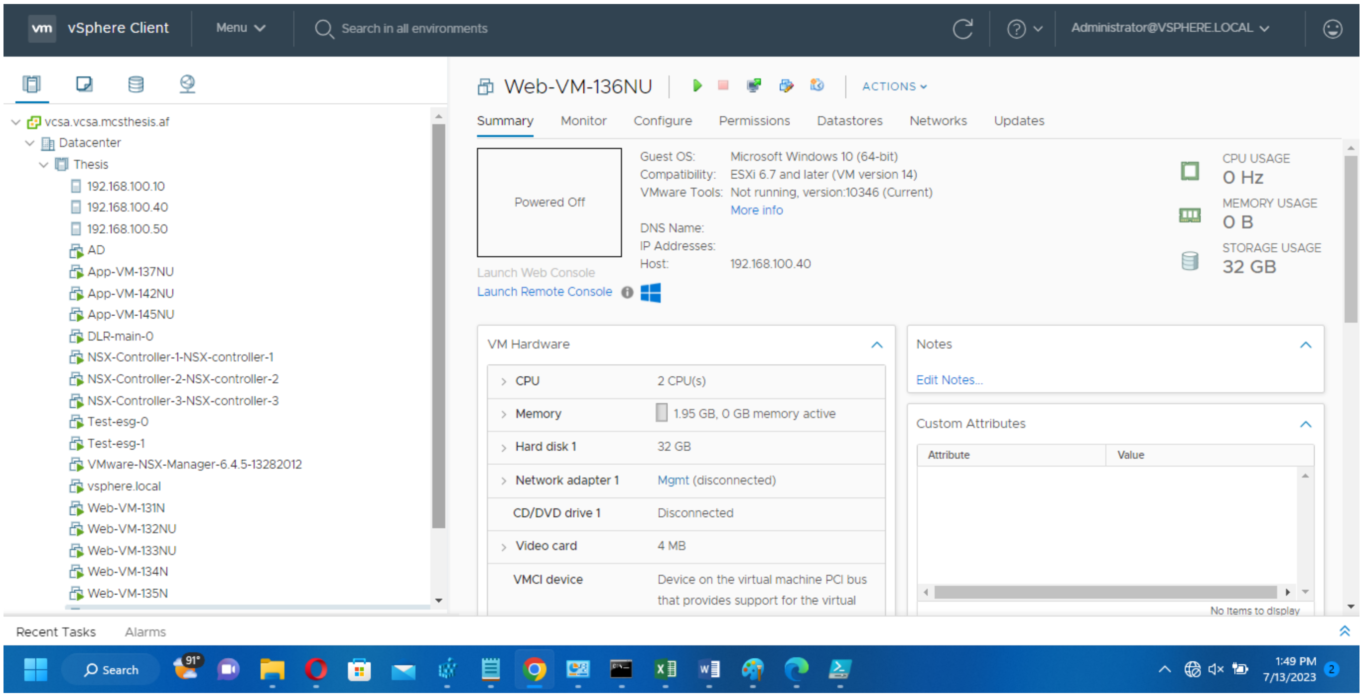
Objects with most alerts 1

| Item | Alerts | Warnings |
|---|---|---|
| vcsa.vcsa.mcsthesis.af | 2 | 1 |

Installed Plugins 4

NSX user interface plugin
VMware vRops Client Plugin

**(Fig 3.3.1 VCSA Environment Summary)**



Web-VM-136NU

Powered Off

Guest OS: Microsoft Windows 10 (64-bit)
Compatibility: ESXi 6.7 and later (VM version 14)
VMware Tools: Not running, version:10346 (Current)
More info
DNS Name:
IP Addresses:
Host: 192.168.100.40

Launch Web Console
Launch Remote Console

CPU USAGE 0 Hz
MEMORY USAGE 0 B
STORAGE USAGE 32 GB

VM Hardware

| | |
|---|---|
| CPU | 2 CPU(s) |
| Memory | 1.95 GB, 0 GB memory active |
| Hard disk 1 | 32 GB |
| Network adapter 1 | Mgmt (disconnected) |
| CD/DVD drive 1 | Disconnected |
| Video card | 4 MB |
| VMCI device | Device on the virtual machine PCI bus that provides support for the virtual |

Notes

Edit Notes...

Custom Attributes

| Attribute | Value |
|---|---|

No items to display

**(Fig 3.3.2 Host and Guest VMs)**

## 3.4 **Proposed VM Scanning and VM Tagging Module**

This module will work as an iterative process that can be customized based on workload, this will automatically scan each VM that has been powered ON and VMware tools have been installed on it. VMware tools are required to communicate with VMs to push script into each VM.

### 3.4.1 **NSX Security Tagging Flags**

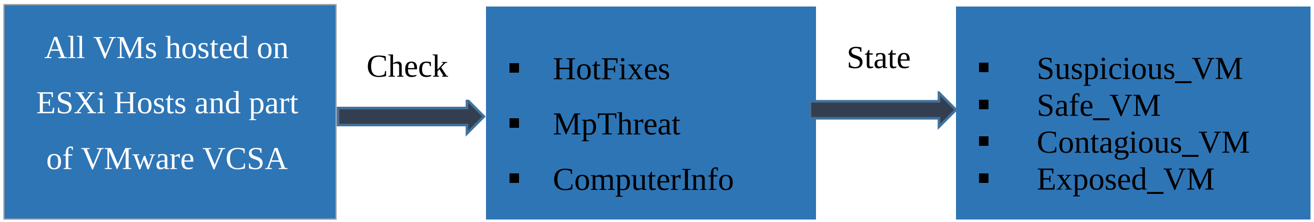
| **Security State** | **Flag** |
|---|---|
| **Protected** | **Safe** |
| **Infected** | **Contagious** |
| **Vulnerable** | **Suspicious** |

## 3.5 **Algorithm 1 VM Scanning Module**

```
Input: [VMs, MPthreat, Hotfixes, ComputerInfo]
Output: [VM Security Status: Contagious VM, Exposed VM, Safe VM, Suspicious VM]
For each VM
    If powered ON AND OS win 10 AND VMware tools installed
        Connect to the VMware environment
        Get a list of all VMs in the environment & Put all VMs in Suspicious State

For each VM in Suspicious State

    Connect to the Suspicious VMs

    Scan VMs to get protection Status

If "active threat count" is 0
        Check the malware definition updates and OS patch updates of the VM
                If both are up-to-date
        Set the VM status as "Safe VM"
          Else
              Set the VM status as "Exposed VM"
     If "active threat count" value is 1 or greater
          Set the VM status as "Contagious VM"
          Else
          "Log Error"
Disconnect from the VM
Export Result of each VM to CSV file
Disconnect from the VMware environment
```

This module's goal is to remotely scan every VM in a VMware environment when a security server in the data center triggers an automated event. To centrally control and carry out the scanning process on all VMs located on the ESXi hosts connected to the VMware vCenter Server Appliance (VCSA), the script establishes a connection with the VCSA. Firstly, it will connect to VCSA so it can manage to run its script from a central console to all VMs that are being hosted on all ESXi hosts part of same VCSA. Initially it will find all VMs in Suspicious state that has been managed through a dynamically set criteria in security groups of VMware NSX. All VMs in any state powered (ON or OFF) and guest OS in windows resultantly will become part of suspicious VMs security group.

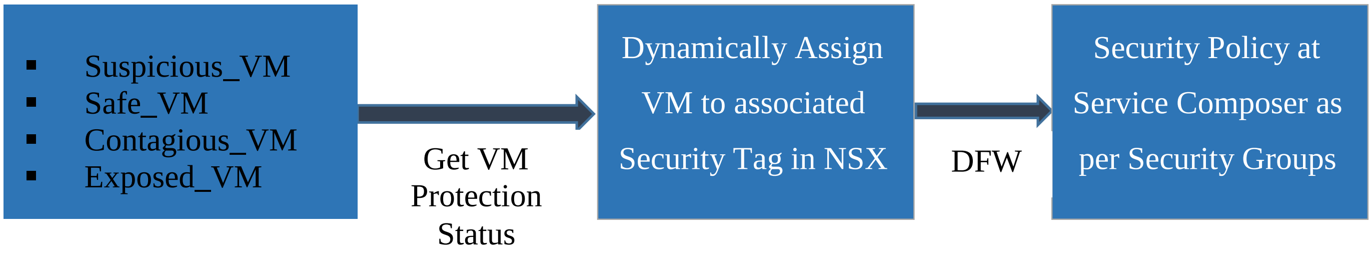
After successfully scanning each VM from suspicious VM security group based on MPthreat and Hotfixes (provide detail malware introspection and OS patch status of VMs) the protection status of each VM will be identified.



**(Fig 3.4.1 VMs Security State Transition)**

## 3.6 NSX DFW Dynamic Security Tag Module

This module will work after successfully getting all VMs protection status. It will take input the protection status of each VM and remotely call the NSX DFW security tag and assign that particular VM against security tag. This process is fully dynamic and iterative VM once assigned against a security tag in NSX will remain assigned to same tag unless protection status of VM remain same. If fresh protection status of VM obtained dynamically security tag will be updated.


▪ Suspicious_VM
▪ Safe_VM
▪ Contagious_VM
▪ Exposed_VM
Get VM Protection Status
Dynamically Assign VM to associated Security Tag in NSX
DFW
Security Policy at Service Composer as per Security Groups

**(Fig 3.6.1 NSX Dynamic Security Tag Assignment)**

### 3.6.1 Algorithm 2 – Dynamically Assigning VM to Security Tag in NSX

```
Input: [VMs Protection Status]
Output: [Dynamically Assign and Re-Assign VM to and from security tag]
Connect to the VMware NSX environment
For each VM from protection status result array
        If protection status = Contagious
                Get a security tag Contagious_VM from NSX security tag
                Assign VM Contagious_VM security tag
        If protection status = Exposed
                Get a security tag Exposed_VM from NSX security tag
                Assign VM Exposed_VM security tag
        If protection status = Safe
                Get a security tag Safe_VM from NSX security tag
                Assign VM Safe_VM security tag
        Else
```

```
                  "Log Error"
Where Security Tag = Safe_VM
        Add VMs assigned in Safe_VM into Safe_VMs_Compartment of NSX security Group
Where Security Tag = Contagious_VM
        Add VMs assigned in Contagious_VM into Contagious_VM compartment of NSX security Group
Where Security Tag = Exposed_VM
        Add VMs assigned in Exposed_VM into Exposed_VM compartment of NSX security Group
Disconnect from the VMware NSX
```

After successfully updating security groups DFW of NSX will create security policies among these groups that will define the lateral movement policy based on defined services. NSX service composer has been used to make custom built services. Lateral policies among security groups are depicted in table 4.2.1.

### 3.6.2 NSX Security Groups

- **Safe_VMs_Compartment = S_VM_C**
- **Exposed_VMs_Compartment = E_VM_C**
- **Contagious_VMs_Compartment = C_VM_C**

### 3.6.3 NSX Service Composer

- **Quarantine_VMs will be applied to E_VM_C**
- **Isolate_VMs will be applied to C_VM_C**
- **Trusted_VMs will be applied to S_VM_C**

| **Service Composer** | **Allowed Services** |
|---|---|
| Isolate_VMs | Deny All |
| Quarantine_VMs | Only minimum desired services<br>Only least privileged access<br>No remote management service |

| | |
|---|---|
| Trusted_VMs | Custom service set<br>Desired remote management services<br>Privileged access rights |

**(Table 3.6.1 Service Composer)**

| **Security Groups** | **S_VM_C** | **E_VMs_C** | **C_VMs_C** |
|---|---|---|---|
| **S_VM_C** | **Trusted_VMs** | Isolate_VMs | Quarantine_VMs |
| **E_VMs_C** | Isolate_VMs | Isolate_VMs | Quarantine_VMs |
| **C_VMs_C** | Quarantine_VMs | Quarantine_VMs | Quarantine_VMs |

**(Table 3.6.2 Security Groups Services Matrix)**

### 3.6.4 Exporting Result Module

This module will export VMs protection status from VM scanning module to CSV file to configured location.

### 3.6.5 Proof of Concept Stack

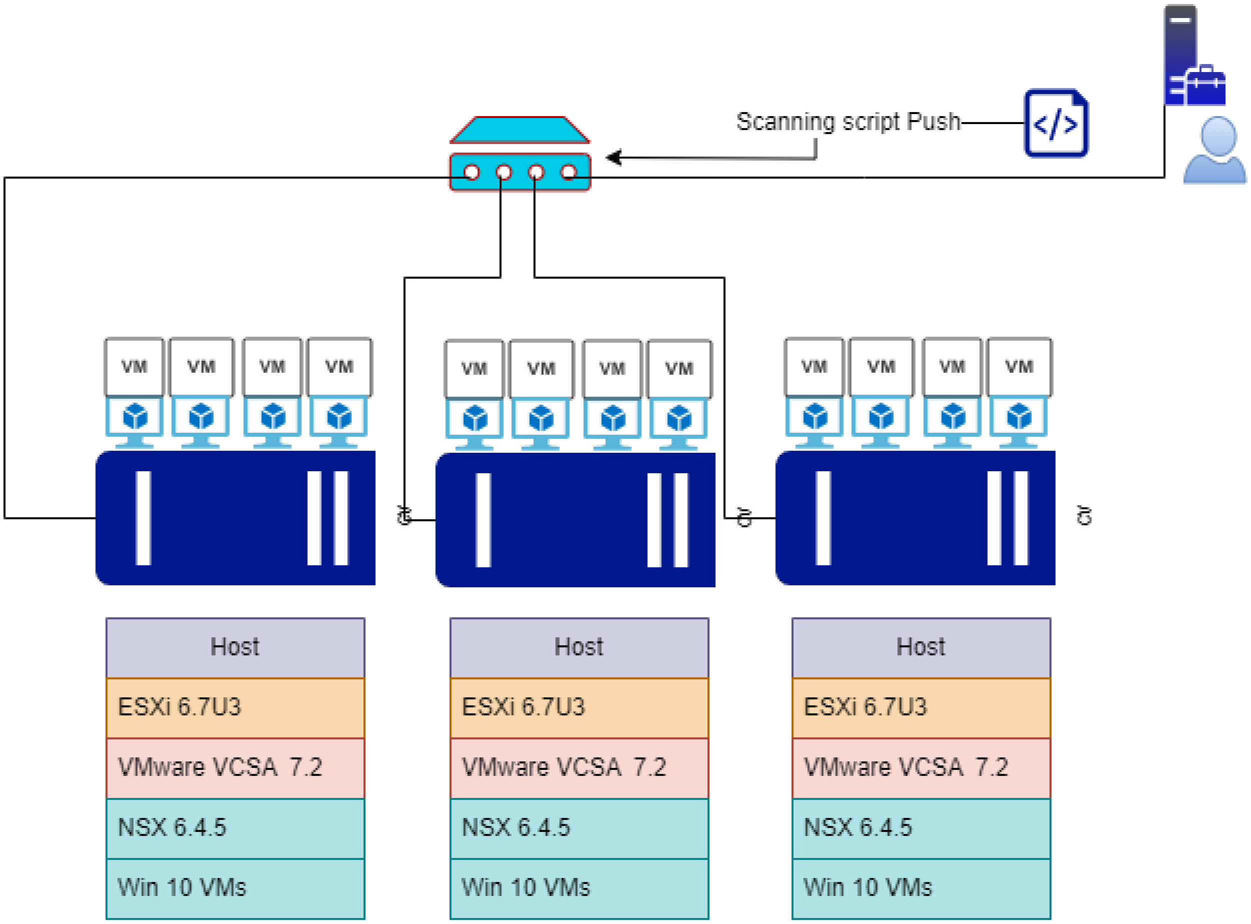
A full-stack virtual data center was set up for a Proof of Concept (PoC) scenario in order to verify the suggested concepts' actual implementation and see how they performed in the real world. Figure 4.4 illustrates the lab deployment paradigm, giving a visual depiction of the architecture and components involved. This PoC deployment's goal was to build a setup that closely matches a production environment so that the suggested concepts and technologies may be thoroughly tested and evaluated. Organizations can learn a lot about the viability, functionality, and efficiency of the deployed solution by simulating a real-world scenario.

All the necessary components for implementing and testing the suggested architecture are included in the lab deployment model (Figure 5.3). It consists of the following elements:

- **ESXi Hosts** Which act as the underlying Type-1 Hypervisors, are included in the deployment. The virtual machines (VMs) within the virtual data center are created and managed by these hosts.

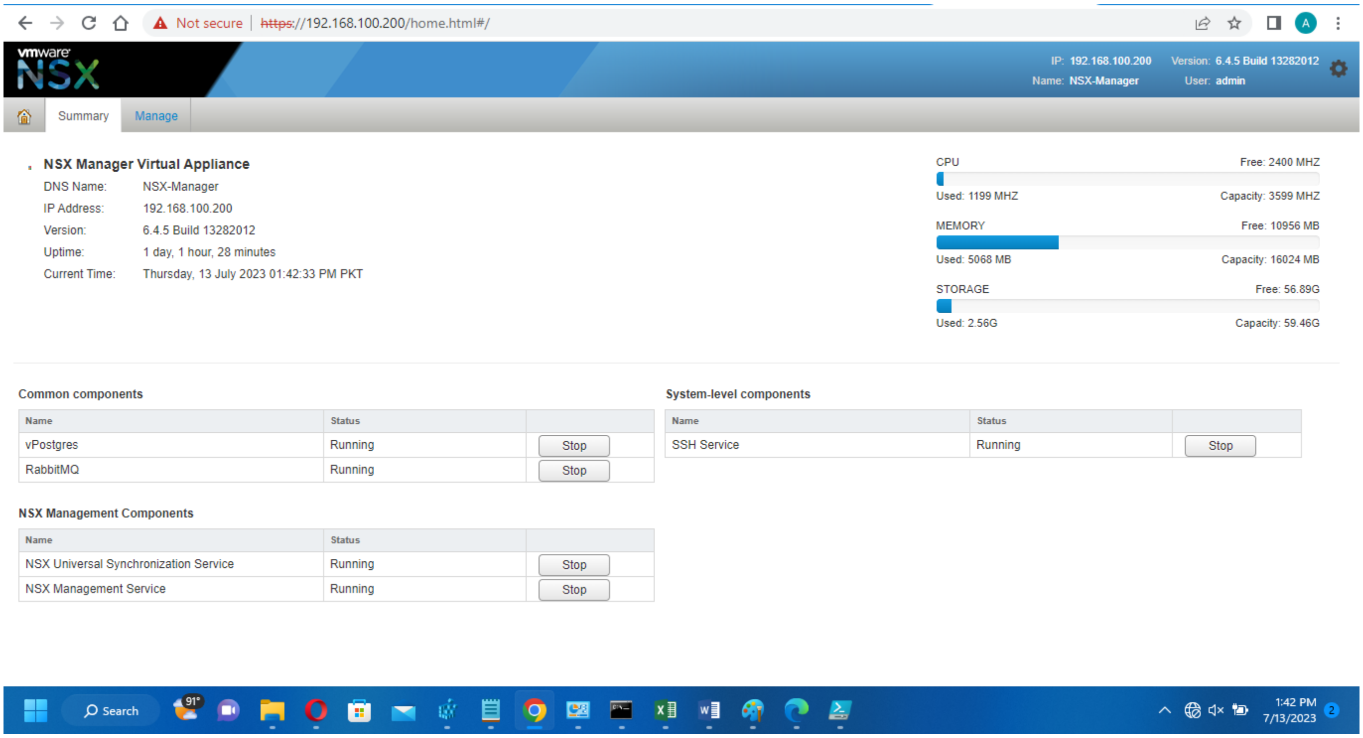
- **VMware vCenter Server Appliance** Centralized control of the virtualized compute resources is provided by the VMware vCenter Server Appliance, a crucial part of the setup. Through a single administration interface, administrators can manage and keep an eye on the ESXi Hosts, VMs, and related resources.
- **Windows 10 VMs** These VMs replicate the various elements and workloads normally present in a production environment, a number of Windows 10 VMs were installed within the virtual data centre. These VMs were dispersed throughout the ESXi Hosts, assuring workload equality and for thorough testing of the suggested remedy.
- **NSX Deployment** A crucial element of the lab setup was the deployment of NSX within the virtual data centre. The network virtualization and security technology, NSX, enables micro-segmentation between the VMs and secure communication. Because of the ability to implement granular access controls and security regulations, the virtual data center's network security is improved.

For testing and evaluating the suggested architecture and concepts, we can say that the lab deployment model shown in Figure 4.4 is thorough and realistic. we can obtain useful information and results that help them fully comprehend the effectiveness and advantages of the suggested solution by simulating a virtual data centre environment and utilizing the capabilities of ESXi Hosts, VMware vCenter Server Appliance, windows 10 VMs, and NSX.



| Host | Host | Host |
|---|---|---|
| ESXi 6.7U3 | ESXi 6.7U3 | ESXi 6.7U3 |
| VMware VCSA 7.2 | VMware VCSA 7.2 | VMware VCSA 7.2 |
| NSX 6.4.5 | NSX 6.4.5 | NSX 6.4.5 |
| Win 10 VMs | Win 10 VMs | Win 10 VMs |

**(Fig 3.6.5 POC Lab Environment)**


Not secure | https://192.168.100.200/home.html#/
vmware NSX
IP: 192.168.100.200 Version: 6.4.5 Build 13282012
Name: NSX-Manager User: admin
Summary
Manage
Search
1:42 PM
7/13/2023

**NSX Manager Virtual Appliance**

| | |
|---|---|
| DNS Name: | NSX-Manager |
| IP Address: | 192.168.100.200 |
| Version: | 6.4.5 Build 13282012 |
| Uptime: | 1 day, 1 hour, 28 minutes |
| Current Time: | Thursday, 13 July 2023 01:42:33 PM PKT |

| | |
|---|---|
| CPU | Free: 2400 MHZ |
| Used: 1199 MHZ | Capacity: 3599 MHZ |
| MEMORY | Free: 10956 MB |
| Used: 5068 MB | Capacity: 16024 MB |
| STORAGE | Free: 56.89G |
| Used: 2.56G | Capacity: 59.46G |

Common components

| Name | Status | |
|---|---|---|
| vPostgres | Running | Stop |
| RabbitMQ | Running | Stop |

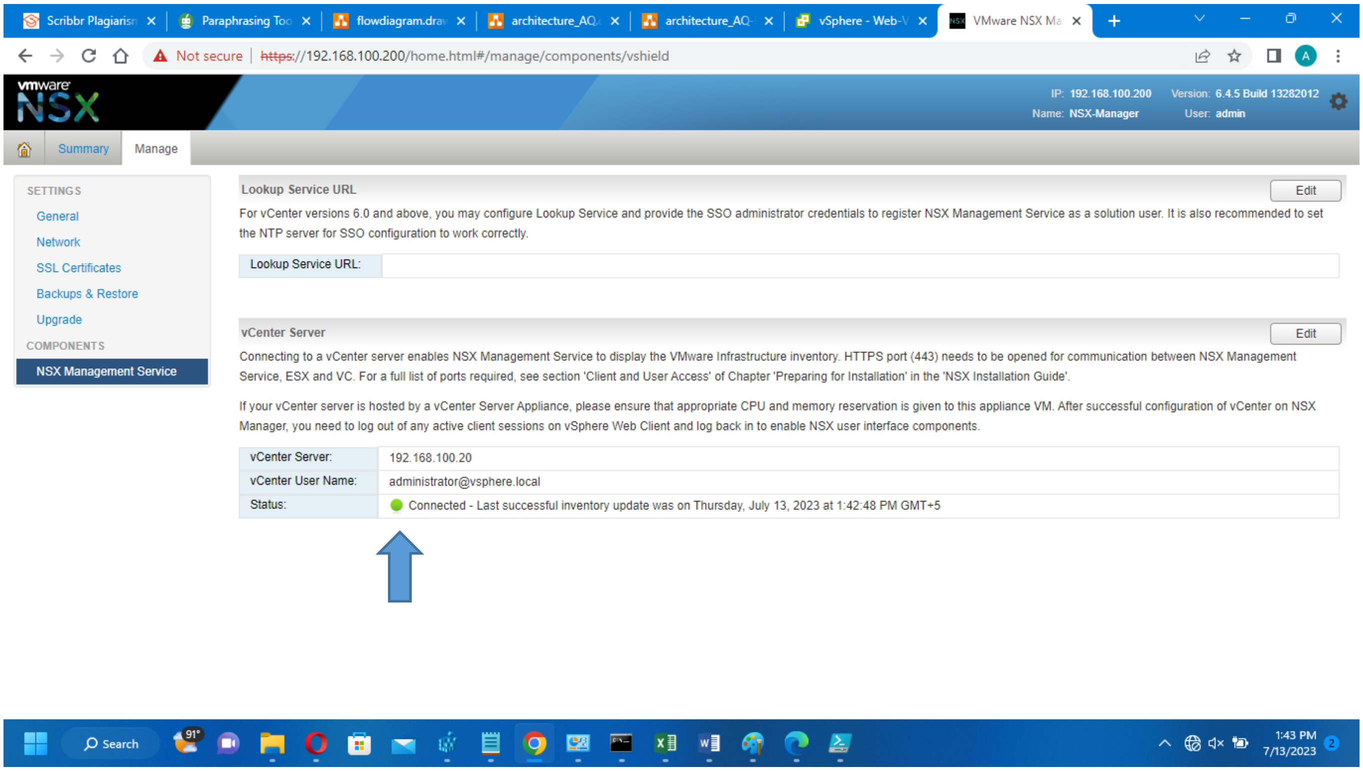
System-level components

| Name | Status | |
|---|---|---|
| SSH Service | Running | Stop |

NSX Management Components

| Name | Status | |
|---|---|---|
| NSX Universal Synchronization Service | Running | Stop |
| NSX Management Service | Running | Stop |

**Fig 3.6.6 NSX Manager Summary**



Lookup Service URL

For vCenter versions 6.0 and above, you may configure Lookup Service and provide the SSO administrator credentials to register NSX Management Service as a solution user. It is also recommended to set the NTP server for SSO configuration to work correctly.

| | |
|---|---|
| Lookup Service URL: | |

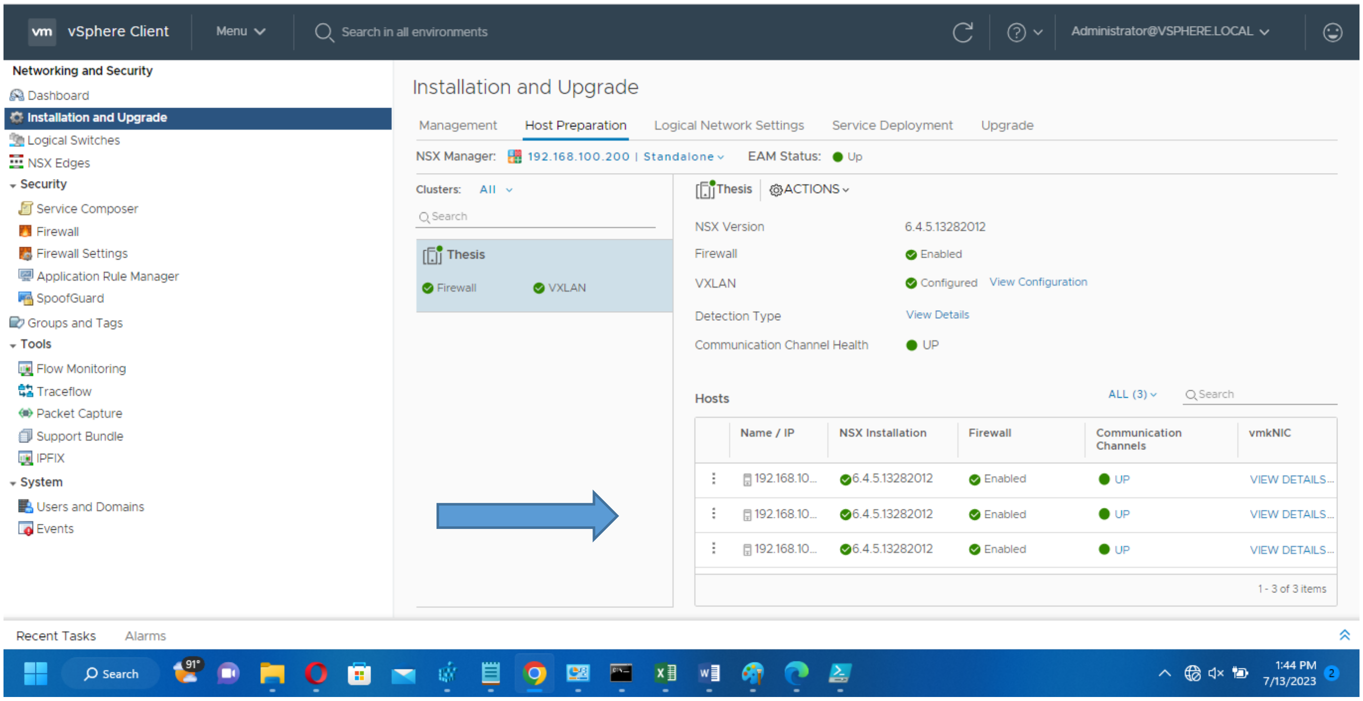
vCenter Server

Connecting to a vCenter server enables NSX Management Service to display the VMware Infrastructure inventory. HTTPS port (443) needs to be opened for communication between NSX Management Service, ESX and VC. For a full list of ports required, see section 'Client and User Access' of Chapter 'Preparing for Installation' in the 'NSX Installation Guide'.

If your vCenter server is hosted by a vCenter Server Appliance, please ensure that appropriate CPU and memory reservation is given to this appliance VM. After successful configuration of vCenter on NSX Manager, you need to log out of any active client sessions on vSphere Web Client and log back in to enable NSX user interface components.

| | |
|---|---|
| vCenter Server: | 192.168.100.20 |
| vCenter User Name: | administrator@vsphere.local |
| Status: | ● Connected - Last successful inventory update was on Thursday, July 13, 2023 at 1:42:48 PM GMT+5 |

**Fig 3.6.7 NSX to VCSA Integration**



NSX Manager: 192.168.100.200 | Standalone

EAM Status: ● Up

| | |
|---|---|
| NSX Version | 6.4.5.13282012 |
| Firewall | ✓ Enabled |
| VXLAN | ✓ Configured View Configuration |
| Detection Type | View Details |
| Communication Channel Health | ● UP |

Hosts ALL (3)

| Name / IP | NSX Installation | Firewall | Communication Channels | vmkNIC |
|---|---|---|---|---|
| 192.168.10... | ✓6.4.5.13282012 | ✓ Enabled | ● UP | VIEW DETAILS... |
| 192.168.10... | ✓6.4.5.13282012 | ✓ Enabled | ● UP | VIEW DETAILS... |
| 192.168.10... | ✓6.4.5.13282012 | ✓ Enabled | ● UP | VIEW DETAILS... |

1 - 3 of 3 items

**Fig 3.6.8 NSX Installed on Hosts**

## 3.7 Flow Diagram of Proposed Methodology

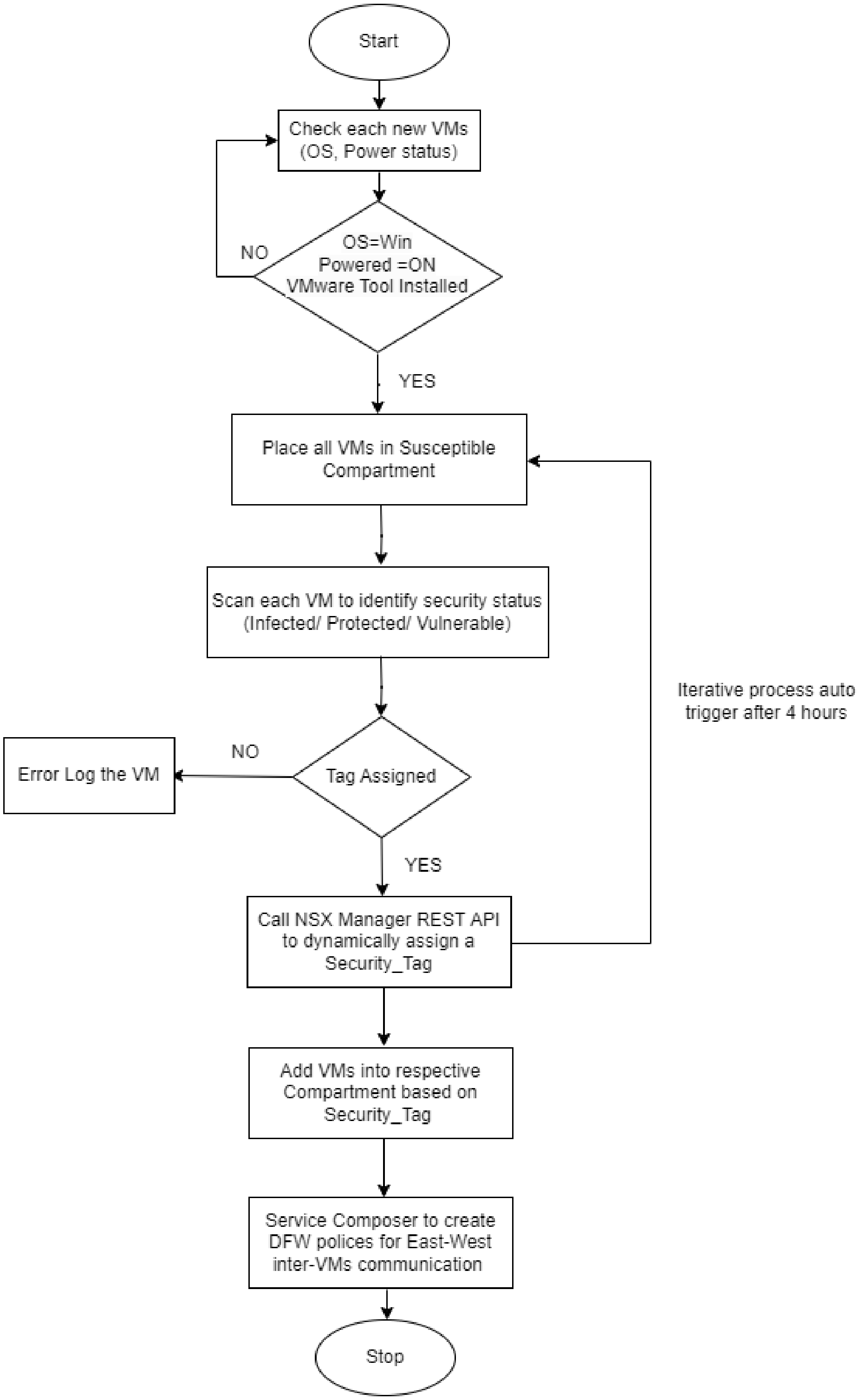

Start
Check each new VMs (OS, Power status)
OS=Win Powered =ON VMware Tool Installed
NO
YES
Place all VMs in Susceptible Compartment
Scan each VM to identify security status (Infected/ Protected/ Vulnerable)
Iterative process auto trigger after 4 hours
Error Log the VM
NO
Tag Assigned
YES
Call NSX Manager REST API to dynamically assign a Security_Tag
Add VMs into respective Compartment based on Security_Tag
Service Composer to create DFW polices for East-West inter-VMs communication
Stop

**Fig 3.7 Flow Chart Proposed Methodology**

*Chapter 4*

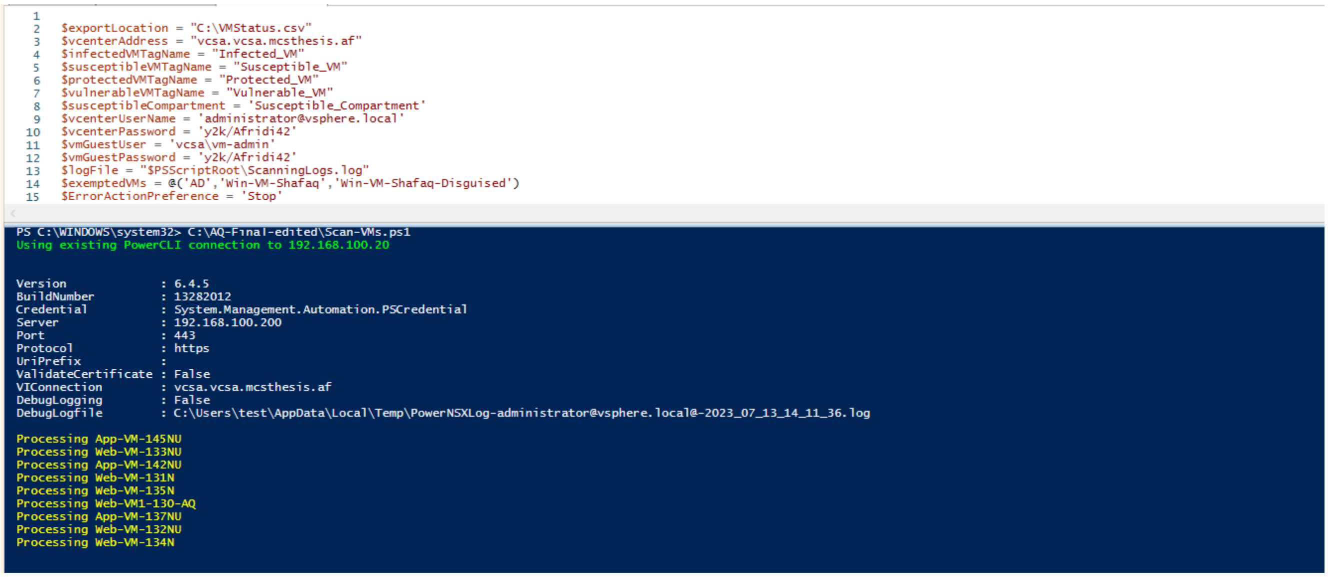
# RESULTS AND ANALYSIS

## 4.1 Phase-1

### 4.1.1 VM Protection Status Scanning

All VMs meeting condition mentioned as per algorithm-1 will be scanned for HotFixes and MPthreatDetection and MPComputerStatus.

**Fig 4.1.1 Machine Status**

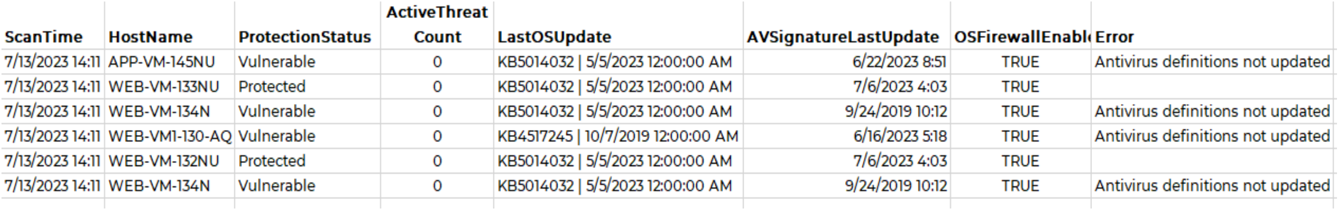


```
$exportLocation = "C:\VMStatus.csv"
$vcenterAddress = "vcsa.vcsa.mcsthesis.af"
$infectedVMTagName = "Infected_VM"
$susceptibleVMTagName = "Susceptible_VM"
$protectedVMTagName = "Protected_VM"
$vulnerableVMTagName = "Vulnerable_VM"
$susceptibleCompartment = 'Susceptible_Compartment'
$vcenterUserName = 'administrator@vsphere.local'
$vcenterPassword = 'y2k/Afridi42'
$vmGuestUser = 'vcsa\vm-admin'
$vmGuestPassword = 'y2k/Afridi42'
$logFile = "$PSScriptRoot\ScanningLogs.log"
$exemptedVMs = @('AD','Win-VM-Shafaq','Win-VM-Shafaq-Disguised')
$ErrorActionPreference = 'Stop'
```

```
PS C:\WINDOWS\system32> C:\AQ-Final-edited\Scan-VMs.ps1
Using existing PowerCLI connection to 192.168.100.20


Version               : 6.4.5
BuildNumber           : 13282012
Credential            : System.Management.Automation.PSCredential
Server                : 192.168.100.200
Port                  : 443
Protocol              : https
UriPrefix             :
ValidateCertificate   : False
VIConnection          : vcsa.vcsa.mcsthesis.af
DebugLogging          : False
DebugLogfile          : C:\Users\test\AppData\Local\Temp\PowerNSXLog-administrator@vsphere.local@-2023_07_13_14_11_36.log

Processing App-VM-145NU
Processing Web-VM-133NU
Processing App-VM-142NU
Processing Web-VM-131N
Processing Web-VM-135N
Processing Web-VM1-130-AQ
Processing App-VM-137NU
Processing Web-VM-132NU
Processing Web-VM-134N
```

**Fig 4.1.2 List VMs Scanned successfully**

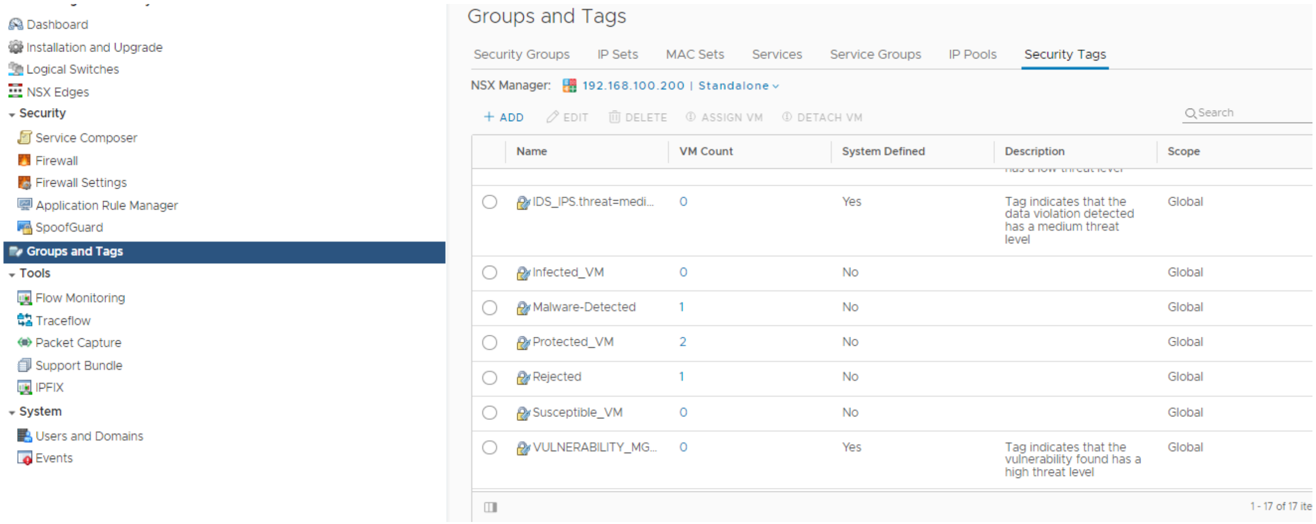
### 4.1.2 VM Protection Status Table (Exported CSV File)



| ScanTime | HostName | ProtectionStatus | ActiveThreat Count | LastOSUpdate | AVSignatureLastUpdate | OSFirewallEnabl | Error |
|---|---|---|---|---|---|---|---|
| 7/13/2023 14:11 | APP-VM-145NU | Vulnerable | 0 | KB5014032 \| 5/5/2023 12:00:00 AM | 6/22/2023 8:51 | TRUE | Antivirus definitions not updated |
| 7/13/2023 14:11 | WEB-VM-133NU | Protected | 0 | KB5014032 \| 5/5/2023 12:00:00 AM | 7/6/2023 4:03 | TRUE | |
| 7/13/2023 14:11 | WEB-VM-134N | Vulnerable | 0 | KB5014032 \| 5/5/2023 12:00:00 AM | 9/24/2019 10:12 | TRUE | Antivirus definitions not updated |
| 7/13/2023 14:11 | WEB-VM1-130-AQ | Vulnerable | 0 | KB4517245 \| 10/7/2019 12:00:00 AM | 6/16/2023 5:18 | TRUE | Antivirus definitions not updated |
| 7/13/2023 14:11 | WEB-VM-132NU | Protected | 0 | KB5014032 \| 5/5/2023 12:00:00 AM | 7/6/2023 4:03 | TRUE | |
| 7/13/2023 14:11 | WEB-VM-134N | Vulnerable | 0 | KB5014032 \| 5/5/2023 12:00:00 AM | 9/24/2019 10:12 | TRUE | Antivirus definitions not updated |

**Table 4.1.2.1 CSV VMs Protection Status**

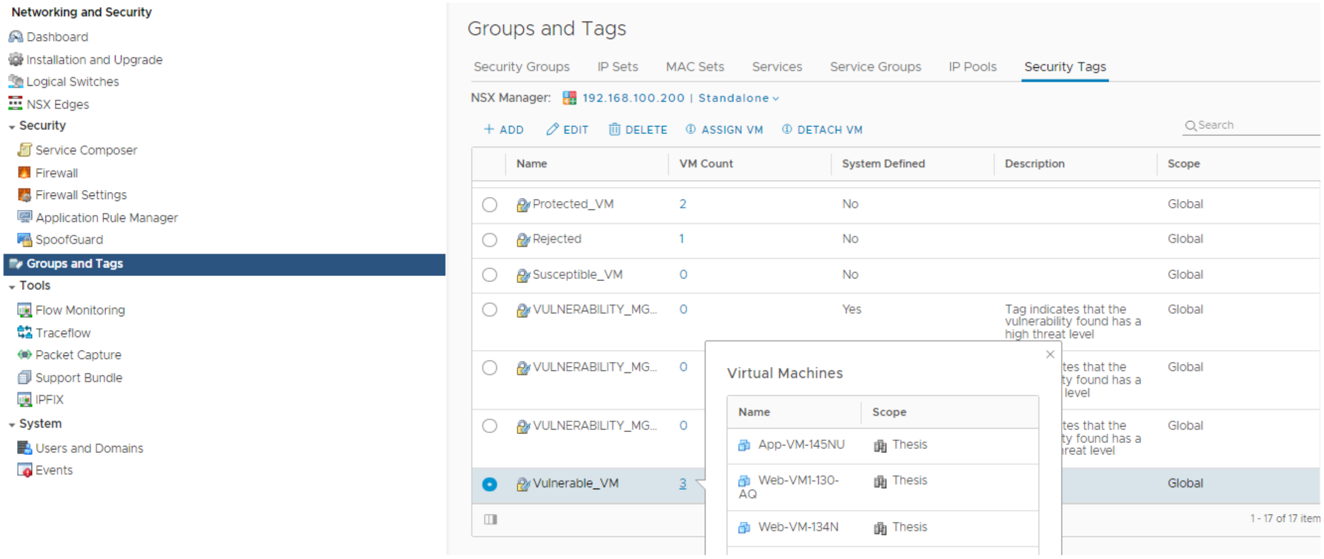
## 4.2 Phase 2 Dynamic Security Tag Allocation in NSX



NSX Manager: 192.168.100.200 | Standalone

| | Name | VM Count | System Defined | Description | Scope |
|---|---|---|---|---|---|
| ○ | IDS_IPS.threat=medi... | 0 | Yes | Tag indicates that the data violation detected has a medium threat level | Global |
| ○ | Infected_VM | 0 | No | | Global |
| ○ | Malware-Detected | 1 | No | | Global |
| ○ | Protected_VM | 2 | No | | Global |
| ○ | Rejected | 1 | No | | Global |
| ○ | Susceptible_VM | 0 | No | | Global |
| ○ | VULNERABILITY_MG... | 0 | Yes | Tag indicates that the vulnerability found has a high threat level | Global |

**Fig 4.2.1 VMs Security Tags Allocation**



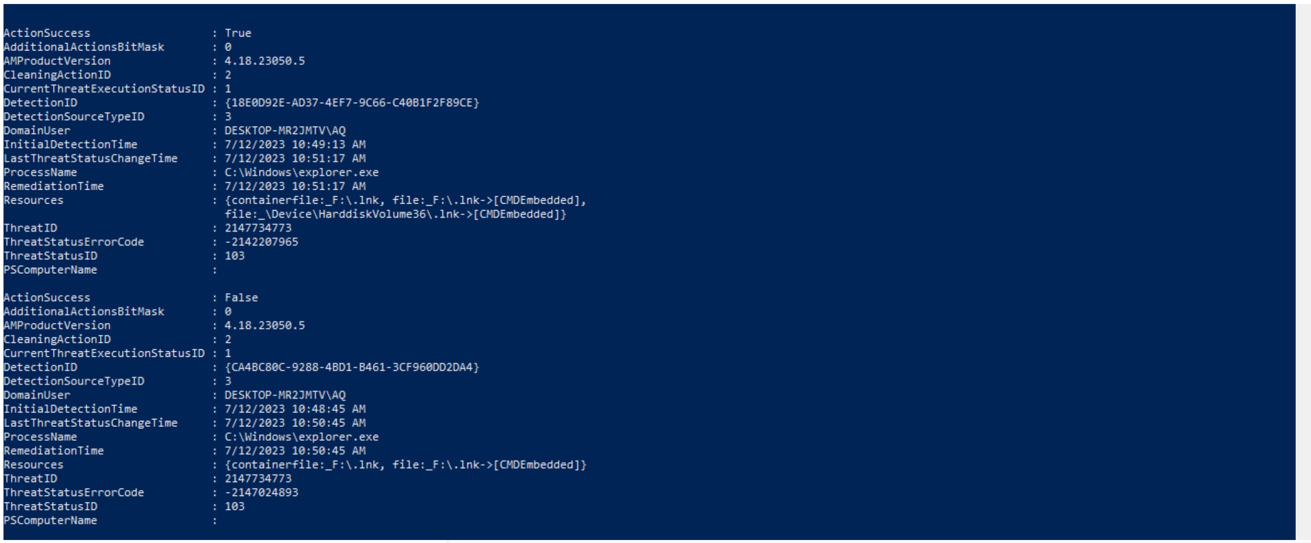
NSX Manager: 192.168.100.200 | Standalone

| | Name | VM Count | System Defined | Description | Scope |
|---|---|---|---|---|---|
| ○ | Protected_VM | 2 | No | | Global |
| ○ | Rejected | 1 | No | | Global |
| ○ | Susceptible_VM | 0 | No | | Global |
| ○ | VULNERABILITY_MG... | 0 | Yes | Tag indicates that the vulnerability found has a high threat level | Global |
| ○ | VULNERABILITY_MG... | 0 | | tes that the ty found has a level | Global |
| ○ | VULNERABILITY_MG... | 0 | | tes that the ty found has a reat level | Global |
| ● | Vulnerable_VM | 3 | | | Global |

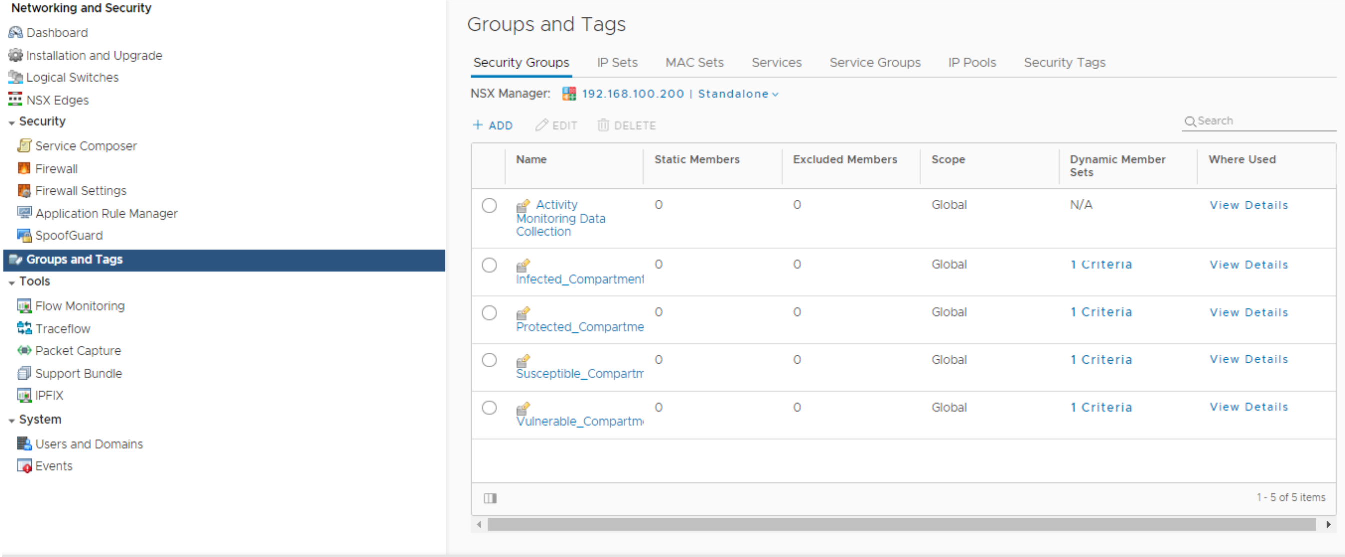
Virtual Machines

| Name | Scope |
|---|---|
| App-VM-145NU | Thesis |
| Web-VM1-130-AQ | Thesis |
| Web-VM-134N | Thesis |

**Fig 4.2.2 VMs Assigned to Associated Security Tag**



```
ActionSuccess                  : True
AdditionalActionsBitMask       : 0
AMProductVersion               : 4.18.23050.5
CleaningActionID               : 2
CurrentThreatExecutionStatusID : 1
DetectionID                    : {18E0D92E-AD37-4EF7-9C66-C40B1F2F89CE}
DetectionSourceTypeID          : 3
DomainUser                     : DESKTOP-MR2JMTV\AQ
InitialDetectionTime           : 7/12/2023 10:49:13 AM
LastThreatStatusChangeTime     : 7/12/2023 10:51:17 AM
ProcessName                    : C:\Windows\explorer.exe
RemediationTime                : 7/12/2023 10:51:17 AM
Resources                      : {containerfile:_F:\.lnk, file:_F:\.lnk->[CMDEmbedded],
                                 file:_\Device\HarddiskVolume36\.lnk->[CMDEmbedded]}
ThreatID                       : 2147734773
ThreatStatusErrorCode          : -2142207965
ThreatStatusID                 : 103
PSComputerName                 :

ActionSuccess                  : False
AdditionalActionsBitMask       : 0
AMProductVersion               : 4.18.23050.5
CleaningActionID               : 2
CurrentThreatExecutionStatusID : 1
DetectionID                    : {CA4BC80C-9288-4BD1-B461-3CF960DD2DA4}
DetectionSourceTypeID          : 3
DomainUser                     : DESKTOP-MR2JMTV\AQ
InitialDetectionTime           : 7/12/2023 10:48:45 AM
LastThreatStatusChangeTime     : 7/12/2023 10:50:45 AM
ProcessName                    : C:\Windows\explorer.exe
RemediationTime                : 7/12/2023 10:50:45 AM
Resources                      : {containerfile:_F:\.lnk, file:_F:\.lnk->[CMDEmbedded]}
ThreatID                       : 2147734773
ThreatStatusErrorCode          : -2147024893
ThreatStatusID                 : 103
PSComputerName                 :
```

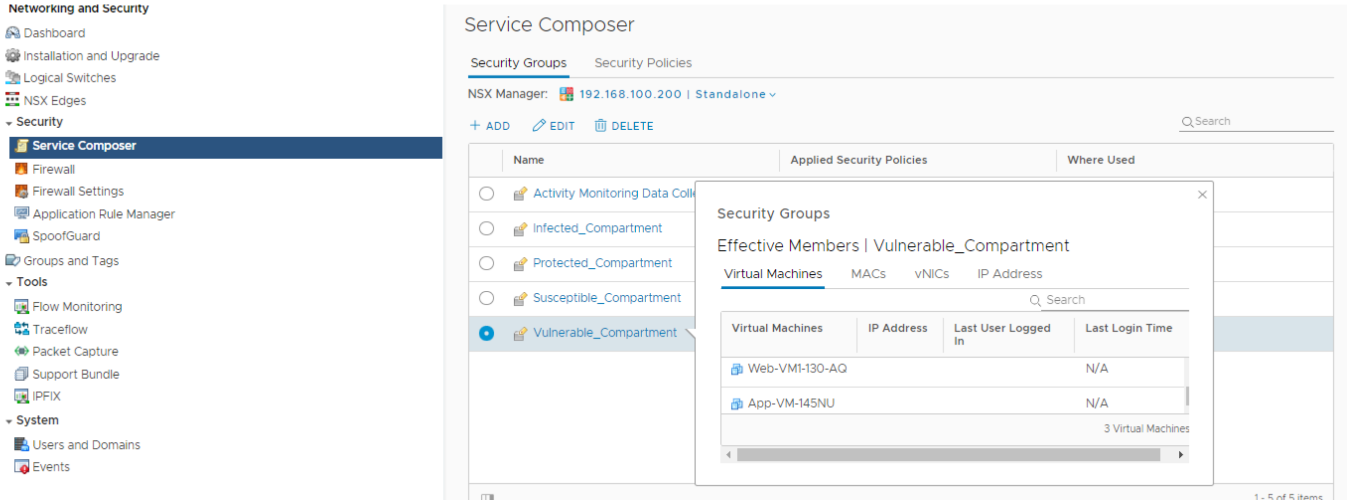


NSX Manager: 192.168.100.200 | Standalone

| | Name | Static Members | Excluded Members | Scope | Dynamic Member Sets | Where Used |
|---|---|---|---|---|---|---|
| ○ | Activity Monitoring Data Collection | 0 | 0 | Global | N/A | View Details |
| ○ | Infected_Compartment | 0 | 0 | Global | 1 Criteria | View Details |
| ○ | Protected_Compartme | 0 | 0 | Global | 1 Criteria | View Details |
| ○ | Susceptible_Compartm | 0 | 0 | Global | 1 Criteria | View Details |
| ○ | Vulnerable_Compartm | 0 | 0 | Global | 1 Criteria | View Details |

1 - 5 of 5 items

## Fig 4.2.3 Security Groups Dynamic Criteria based Assignment of Security Tags



NSX Manager: 192.168.100.200 | Standalone

| | Name | Applied Security Policies | Where Used |
|---|---|---|---|
| ○ | Activity Monitoring Data Coll | | |
| ○ | Infected_Compartment | | |
| ○ | Protected_Compartment | | |
| ○ | Susceptible_Compartment | | |
| ● | Vulnerable_Compartment | | |

Security Groups

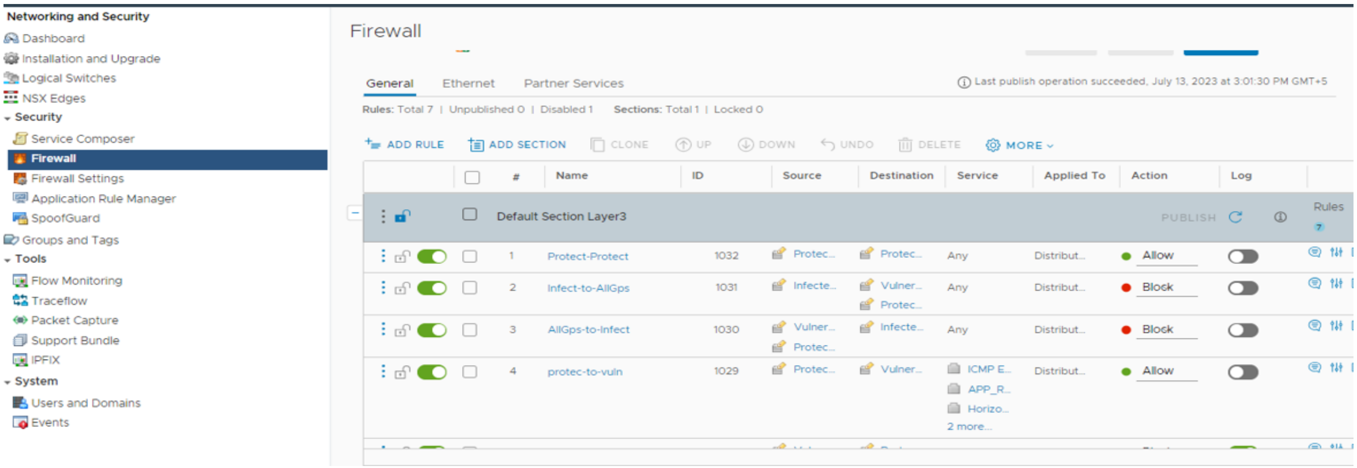
Effective Members | Vulnerable_Compartment

| Virtual Machines | IP Address | Last User Logged In | Last Login Time |
|---|---|---|---|
| Web-VM1-130-AQ | | | N/A |
| App-VM-145NU | | | N/A |

3 Virtual Machines

1 - 5 of 5 items

**<u>Fig 4.2.4 Service Composer Dynamic Collection of VM into a Single Container</u>**



Last publish operation succeeded, July 13, 2023 at 3:01:30 PM GMT+5

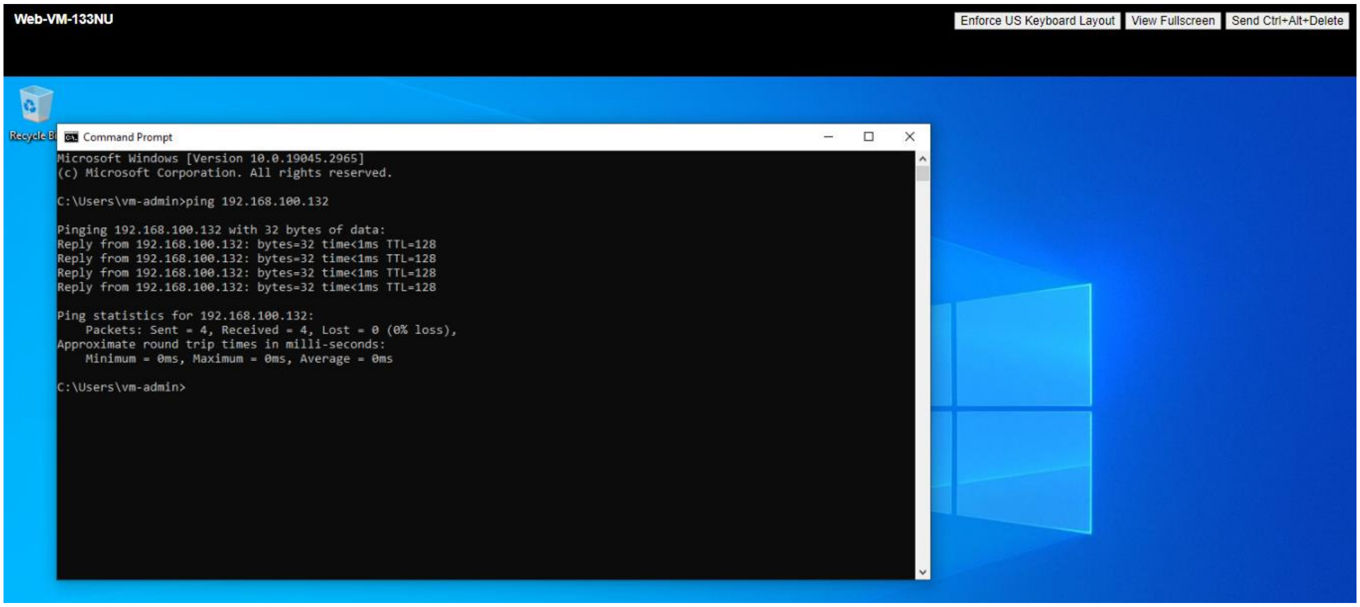
Rules: Total 7 | Unpublished 0 | Disabled 1   Sections: Total 1 | Locked 0

Default Section Layer3 — Rules 7

| # | Name | ID | Source | Destination | Service | Applied To | Action | Log |
|---|---|---|---|---|---|---|---|---|
| 1 | Protect-Protect | 1032 | Protec... | Protec... | Any | Distribut... | Allow | |
| 2 | Infect-to-AllGps | 1031 | Infecte... | Vulner...<br>Protec... | Any | Distribut... | Block | |
| 3 | AllGps-to-Infect | 1030 | Vulner...<br>Protec... | Infecte... | Any | Distribut... | Block | |
| 4 | protec-to-vuln | 1029 | Protec... | Vulner... | ICMP E...<br>APP_R...<br>Horizo...<br>2 more... | Distribut... | Allow | |

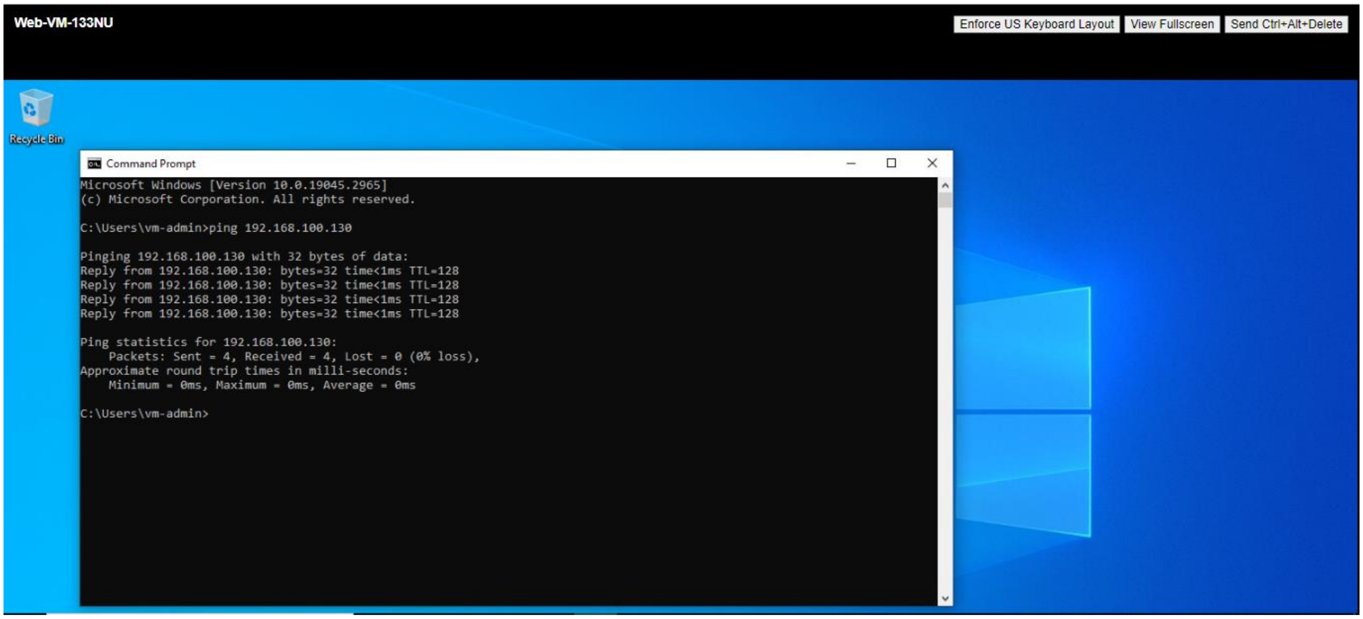
**<u>Fig 4.2.5 DFW Rules applied to Security Groups as per Matrix</u>**

4.3 **Lateral Traffic Limiting**



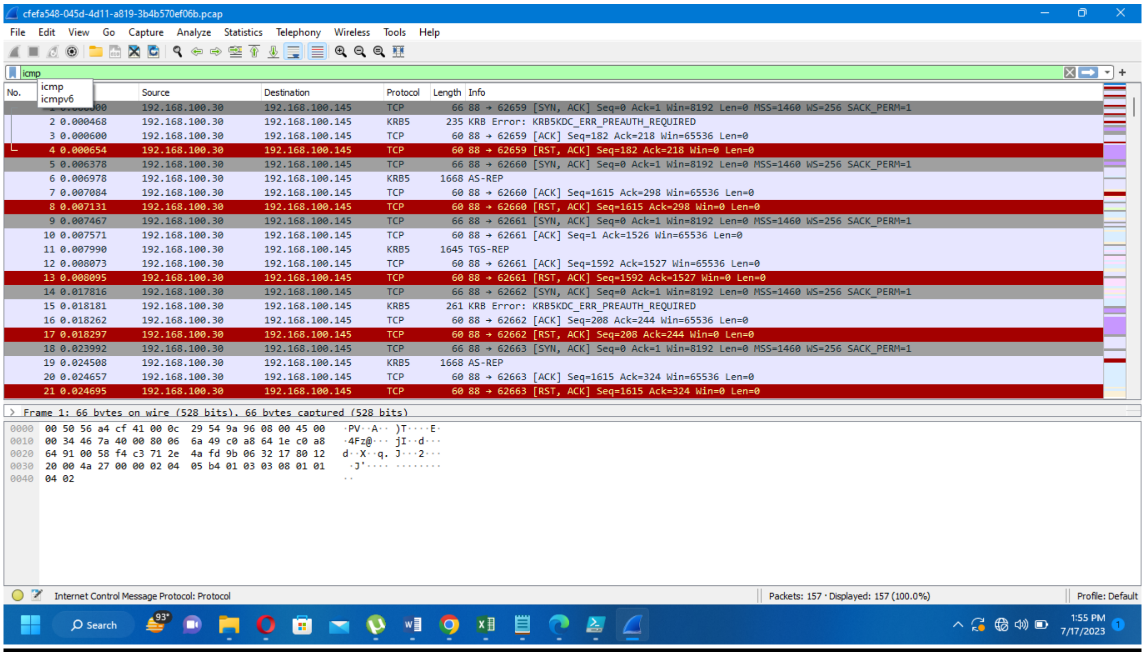
```
Microsoft Windows [Version 10.0.19045.2965]
(c) Microsoft Corporation. All rights reserved.

C:\Users\vm-admin>ping 192.168.100.132

Pinging 192.168.100.132 with 32 bytes of data:
Reply from 192.168.100.132: bytes=32 time<1ms TTL=128
Reply from 192.168.100.132: bytes=32 time<1ms TTL=128
Reply from 192.168.100.132: bytes=32 time<1ms TTL=128
Reply from 192.168.100.132: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.100.132:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\vm-admin>
```

**Fig 4.3.1 ICMP Status Protected_VM to Protected_VM**



```
Microsoft Windows [Version 10.0.19045.2965]
(c) Microsoft Corporation. All rights reserved.

C:\Users\vm-admin>ping 192.168.100.130

Pinging 192.168.100.130 with 32 bytes of data:
Reply from 192.168.100.130: bytes=32 time<1ms TTL=128
Reply from 192.168.100.130: bytes=32 time<1ms TTL=128
Reply from 192.168.100.130: bytes=32 time<1ms TTL=128
Reply from 192.168.100.130: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.100.130:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\vm-admin>
```

**Fig 4.3.2 ICMP Status Protected_VM to Vulnerable_VM**

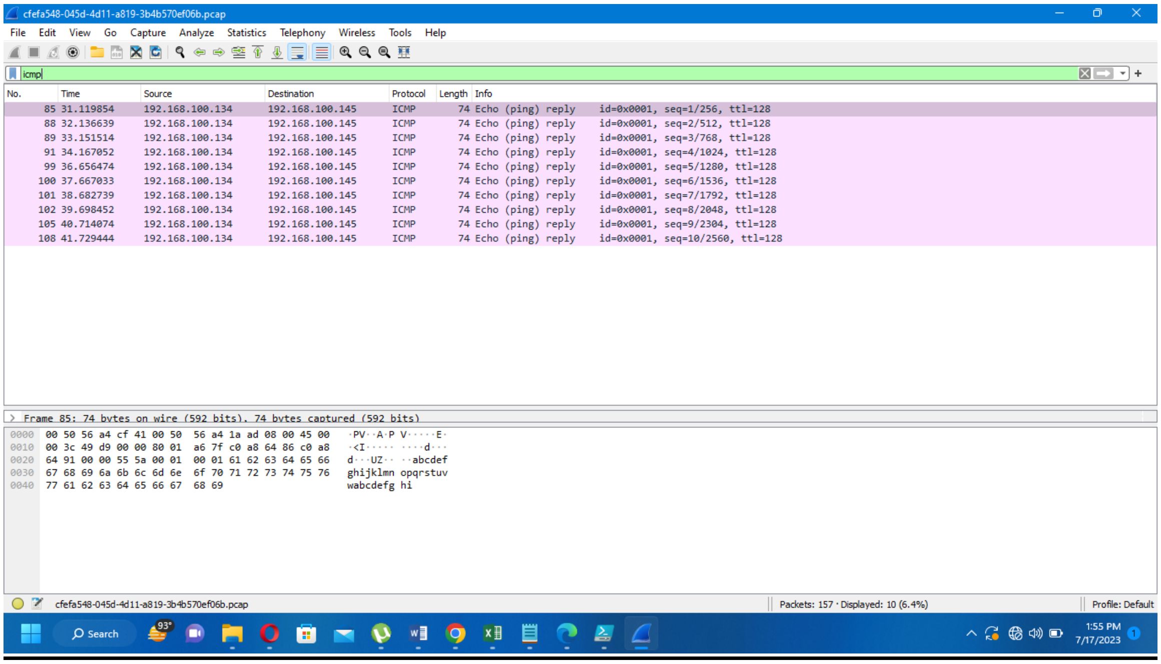

cfefa548-045d-4d11-a819-3b4b570ef06b.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmp
icmpv6

Filter: icmp

```
No.  Time       Source          Destination      Protocol Length Info
   1 0.000000   192.168.100.30  192.168.100.145  TCP        66 88 → 62659 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
   2 0.000468   192.168.100.30  192.168.100.145  KRB5      235 KRB Error: KRB5KDC_ERR_PREAUTH_REQUIRED
   3 0.000600   192.168.100.30  192.168.100.145  TCP        60 88 → 62659 [ACK] Seq=182 Ack=218 Win=65536 Len=0
   4 0.000654   192.168.100.30  192.168.100.145  TCP        60 88 → 62659 [RST, ACK] Seq=182 Ack=218 Win=0 Len=0
   5 0.006378   192.168.100.30  192.168.100.145  TCP        66 88 → 62660 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
   6 0.006978   192.168.100.30  192.168.100.145  KRB5     1668 AS-REP
   7 0.007084   192.168.100.30  192.168.100.145  TCP        60 88 → 62660 [ACK] Seq=1615 Ack=298 Win=65536 Len=0
   8 0.007131   192.168.100.30  192.168.100.145  TCP        60 88 → 62660 [RST, ACK] Seq=1615 Ack=298 Win=0 Len=0
   9 0.007467   192.168.100.30  192.168.100.145  TCP        66 88 → 62661 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
  10 0.007571   192.168.100.30  192.168.100.145  TCP        60 88 → 62661 [ACK] Seq=1 Ack=1526 Win=65536 Len=0
  11 0.007990   192.168.100.30  192.168.100.145  KRB5     1645 TGS-REP
  12 0.008073   192.168.100.30  192.168.100.145  TCP        60 88 → 62661 [ACK] Seq=1592 Ack=1527 Win=65536 Len=0
  13 0.008095   192.168.100.30  192.168.100.145  TCP        60 88 → 62661 [RST, ACK] Seq=1592 Ack=1527 Win=0 Len=0
  14 0.017816   192.168.100.30  192.168.100.145  TCP        66 88 → 62662 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
  15 0.018181   192.168.100.30  192.168.100.145  KRB5      261 KRB Error: KRB5KDC_ERR_PREAUTH_REQUIRED
  16 0.018262   192.168.100.30  192.168.100.145  TCP        60 88 → 62662 [ACK] Seq=208 Ack=244 Win=65536 Len=0
  17 0.018297   192.168.100.30  192.168.100.145  TCP        60 88 → 62662 [RST, ACK] Seq=208 Ack=244 Win=0 Len=0
  18 0.023992   192.168.100.30  192.168.100.145  TCP        66 88 → 62663 [SYN, ACK] Seq=0 Ack=1 Win=8192 Len=0 MSS=1460 WS=256 SACK_PERM=1
  19 0.024508   192.168.100.30  192.168.100.145  KRB5     1668 AS-REP
  20 0.024657   192.168.100.30  192.168.100.145  TCP        60 88 → 62663 [ACK] Seq=1615 Ack=324 Win=65536 Len=0
  21 0.024695   192.168.100.30  192.168.100.145  TCP        60 88 → 62663 [RST, ACK] Seq=1615 Ack=324 Win=0 Len=0
```

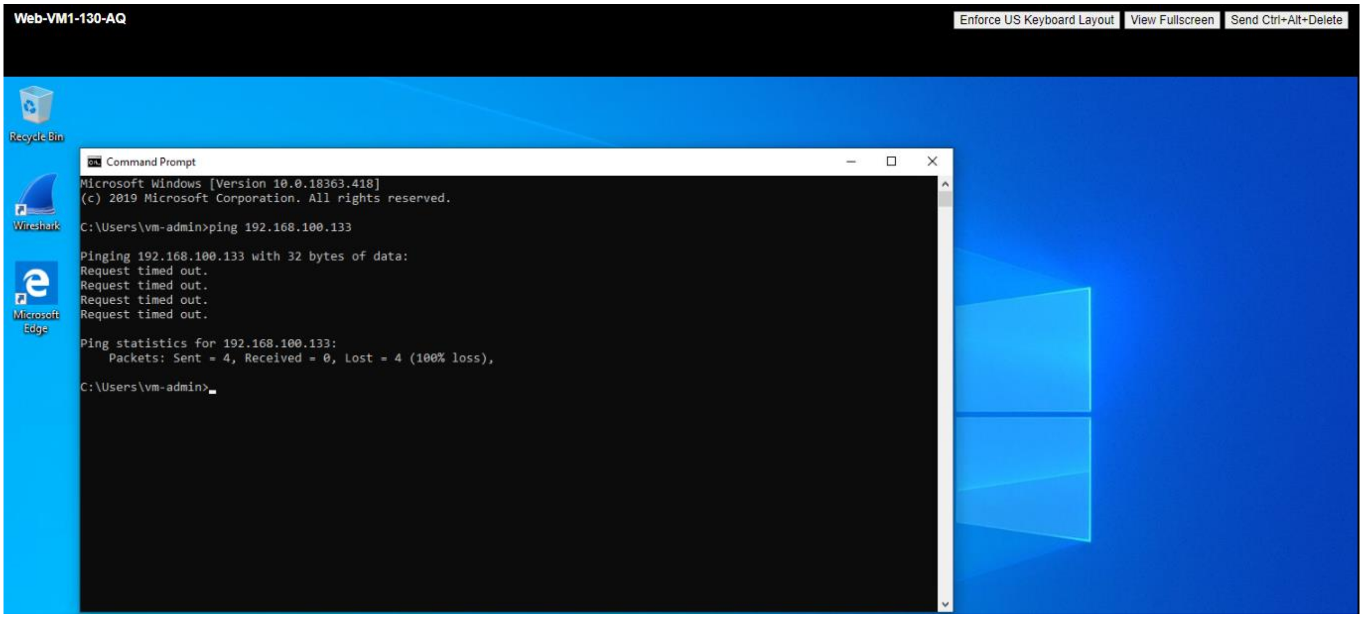
```
> Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits)

0000  00 50 56 a4 cf 41 00 0c  29 54 9a 96 08 00 45 00   ·PV··A·· )T····E·
0010  00 34 46 7a 40 00 80 06  6a 49 c0 a8 64 1e c0 a8   ·4Fz@··· jI··d···
0020  64 91 00 58 f4 c3 71 2e  4a fd 9b 06 32 17 80 12   d··X··q. J···2···
0030  20 00 4a 27 00 00 02 04  05 b4 01 03 03 08 01 01    ·J'···· ········
0040  04 02                                              ··
```

Internet Control Message Protocol: Protocol | Packets: 157 · Displayed: 157 (100.0%) | Profile: Default


cfefa548-045d-4d11-a819-3b4b570ef06b.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmp
Search
93°
1:55 PM
7/17/2023

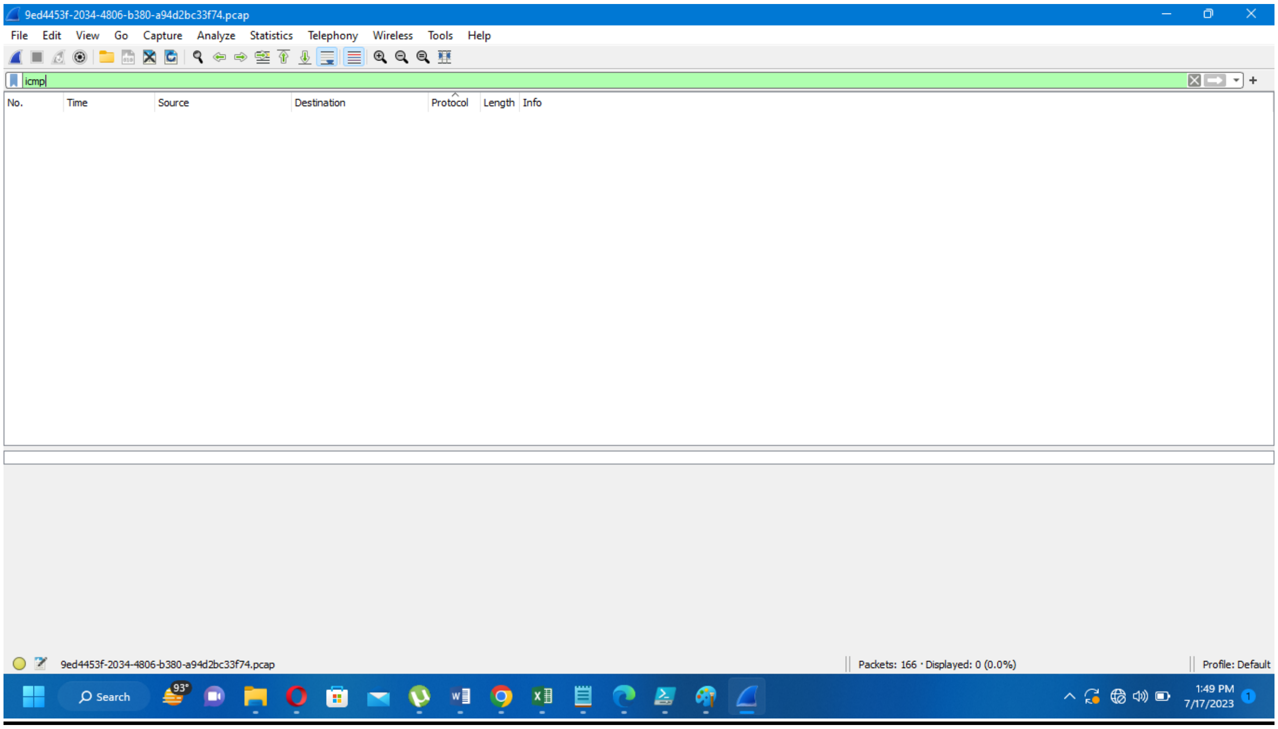
```
No.    Time        Source            Destination       Protocol Length Info
    85 31.119854   192.168.100.134   192.168.100.145   ICMP       74 Echo (ping) reply    id=0x0001, seq=1/256, ttl=128
    88 32.136639   192.168.100.134   192.168.100.145   ICMP       74 Echo (ping) reply    id=0x0001, seq=2/512, ttl=128
    89 33.151514   192.168.100.134   192.168.100.145   ICMP       74 Echo (ping) reply    id=0x0001, seq=3/768, ttl=128
    91 34.167052   192.168.100.134   192.168.100.145   ICMP       74 Echo (ping) reply    id=0x0001, seq=4/1024, ttl=128
    99 36.656474   192.168.100.134   192.168.100.145   ICMP       74 Echo (ping) reply    id=0x0001, seq=5/1280, ttl=128
   100 37.667033   192.168.100.134   192.168.100.145   ICMP       74 Echo (ping) reply    id=0x0001, seq=6/1536, ttl=128
   101 38.682739   192.168.100.134   192.168.100.145   ICMP       74 Echo (ping) reply    id=0x0001, seq=7/1792, ttl=128
   102 39.698452   192.168.100.134   192.168.100.145   ICMP       74 Echo (ping) reply    id=0x0001, seq=8/2048, ttl=128
   105 40.714074   192.168.100.134   192.168.100.145   ICMP       74 Echo (ping) reply    id=0x0001, seq=9/2304, ttl=128
   108 41.729444   192.168.100.134   192.168.100.145   ICMP       74 Echo (ping) reply    id=0x0001, seq=10/2560, ttl=128

> Frame 85: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)

0000  00 50 56 a4 cf 41 00 50  56 a4 1a ad 08 00 45 00   ·PV··A·P V·····E·
0010  00 3c 49 d9 00 00 80 01  a6 7f c0 a8 64 86 c0 a8   ·<I····· ····d···
0020  64 91 00 00 55 5a 00 01  00 01 61 62 63 64 65 66   d···UZ·· ··abcdef
0030  67 68 69 6a 6b 6c 6d 6e  6f 70 71 72 73 74 75 76   ghijklmn opqrstuv
0040  77 61 62 63 64 65 66 67  68 69                     wabcdefg hi

cfefa548-045d-4d11-a819-3b4b570ef06b.pcap    Packets: 157 · Displayed: 10 (6.4%)    Profile: Default
```

## Fig 4.3.2a ICMP Wireshark Status Protected_VM to Vulnerable_VM



```
Microsoft Windows [Version 10.0.18363.418]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\vm-admin>ping 192.168.100.133

Pinging 192.168.100.133 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.100.133:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\Users\vm-admin>_
```

## Fig 4.3.3 ICMP Status Vulnerable_VM to Protected_VM


File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Search
1:49 PM
7/17/2023

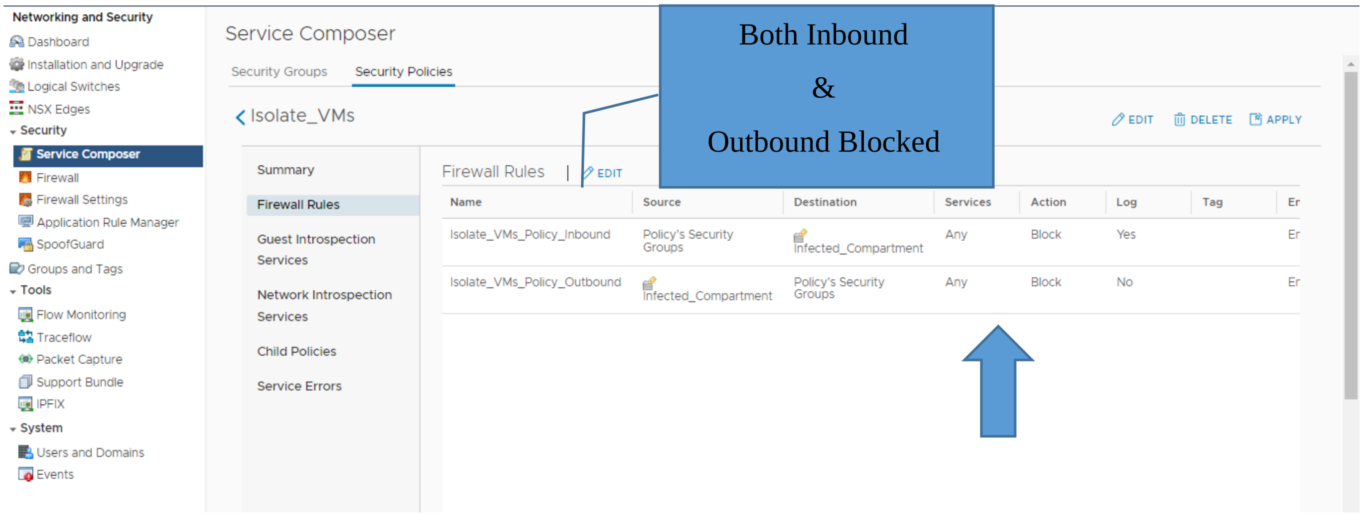
9ed4453f-2034-4806-b380-a94d2bc33f74.pcap

icmp

No. | Time | Source | Destination | Protocol | Length | Info

9ed4453f-2034-4806-b380-a94d2bc33f74.pcap | Packets: 166 · Displayed: 0 (0.0%) | Profile: Default

**Fig 4.3.3a ICMP Nil Wireshark Status Vulnerable_VM to Protected_VM**



Both Inbound

&

Outbound Blocked

Firewall Rules

| Name | Source | Destination | Services | Action | Log | Tag | Er |
|---|---|---|---|---|---|---|---|
| Isolate_VMs_Policy_Inbound | Policy's Security Groups | Infected_Compartment | Any | Block | Yes | | Er |
| Isolate_VMs_Policy_Outbound | Infected_Compartment | Policy's Security Groups | Any | Block | No | | Er |

**Fig 4.3.4 Isolated VMs Policy Status (Infected_VMs)**

## 4.4 Results Analysis

4.4.1 **Micro-segmentation** The concept of micro-segmenting a network involves breaking it into smaller, isolated zones or segments, often at the application or workload level. It enables for accurate access control and security regulations and offers granular control over network traffic.

4.4.2 **Network segmentation** Depending on criteria like departments, functions, or security considerations, a network may be segmented into several logical segments or subnetworks. It establishes distinct network segments for various device or user groups.

4.4.3 **Security Effectiveness**

4.4.3.1 **Malware Contained** Measure the containment rate of malware within the segmented areas for both micro-segmentation and network segmentation approaches.

| **Per Segment/workload Efficiency** | **Parameters** | | |
|---|---|---|---|
| | **No of Malware Threats Detects** | **Malware Incidents Contained** | **%Age of Malware Incidents Contained** |
| **Network Segment** | 5 | 1 | 20% |
| **Micro-Segment** | 5 | 4* | 80% |

*used privilege escalation to copy later adjusted through domain policy

**Table 4.4.3.1 Malware Contained**

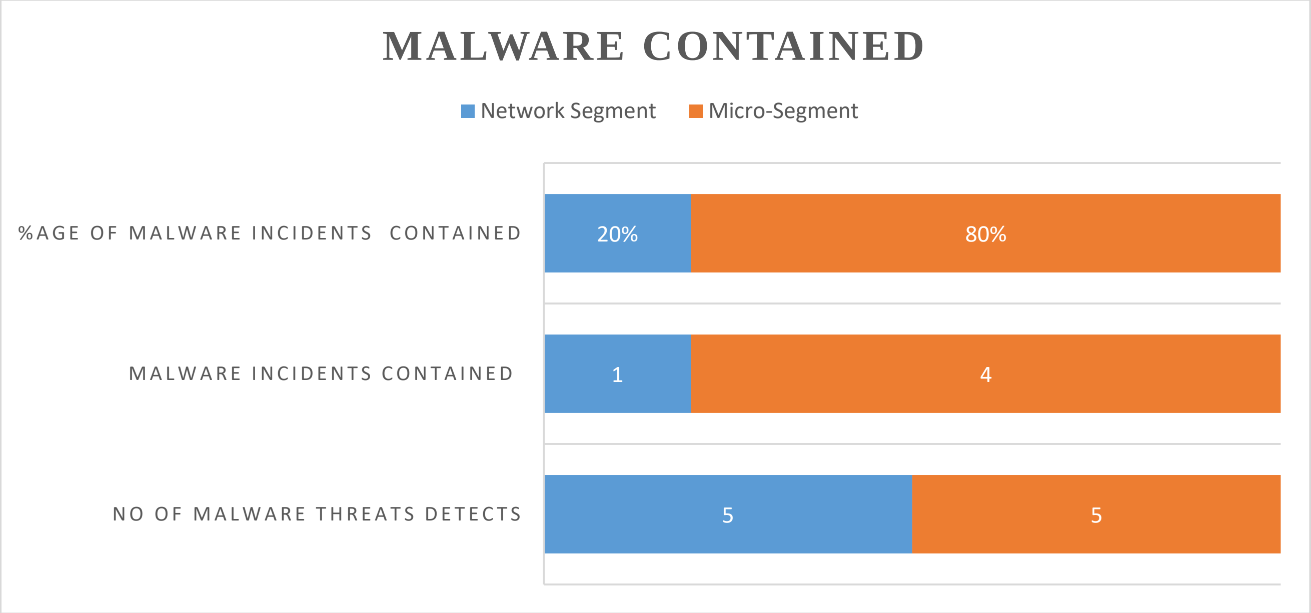


**Fig 4.4.1 Malware Contained in Network Segment Vs Micro-Segment**

4.4.4 **Lateral Movement** Analyze the ability of malware to spread laterally within the network segments and measure the success of containment.

| **Per Segment/workload Efficiency** | **Parameters** | | |
|---|---|---|---|
| | **No of Malware Threats Detects** | **Malware Incidents Contained** | **%Age of Malware Incidents Propagated** |
| **Network Segment Layer-3** | 5 | 3* | 60% |
| **Network Segments Layer-2** | 5 | 5** | 100% |
| **Micro-Segment** | 5 | 0*** | 0% |

*common gateway **Same VLAN ***Isolated Security Group

**Table 4.4.3.2 Malware Propagation**

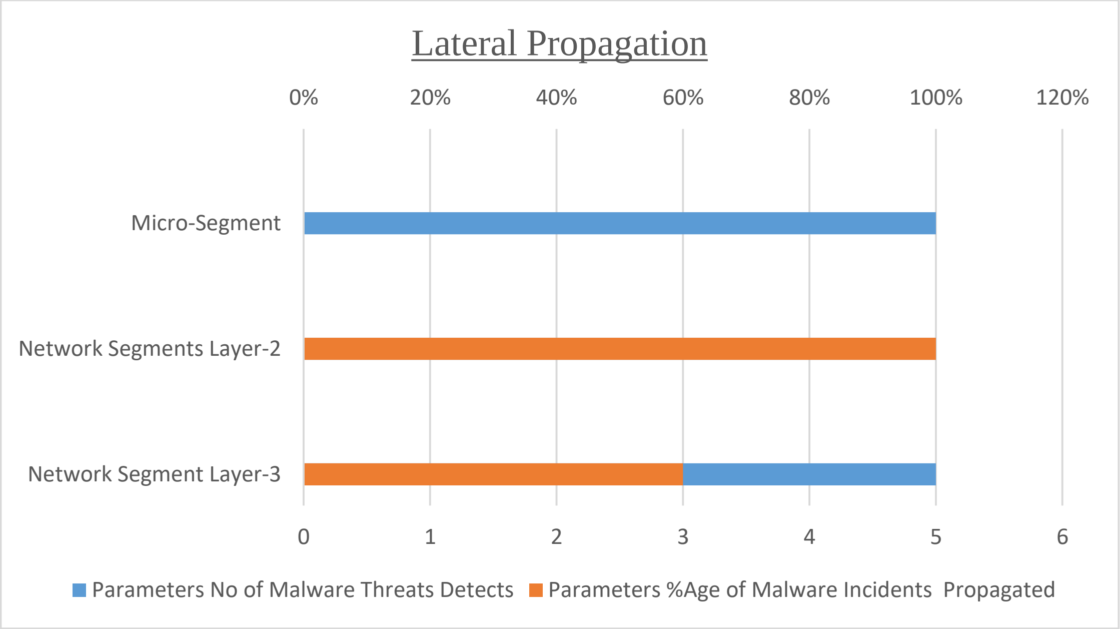


**Fig 4.4.1 Lateral Propagation Analysis Chart**

### 4.4.5 Implementation of Security Controls

| **Segments** | **Security Controls** | | | | |
|---|---|---|---|---|---|
| | **IDS/IPS** | **Anti-Virus Software** | **Threat Intelligence Feed** | **Traffic Monitoring Tools** | **Traffic Analysis Tool** |
| **Network Segment** | No | Windows Defender | No | Wireshark | Wireshark |
| **Micro-Segment** | No | Windows Defender | No | NSX Traffic Capture | Wireshark |

**Table 4.4.5 Implementation of Controls**

### 4.4.6 Collection of Incident Data

| **Segments** | **Incident Data Collection** | | | | |
|---|---|---|---|---|---|
| | **Security Logs** | **SIEM** | **Antivirus** | **Traffic Monitoring Tools** | **Collection Tool** |
| **Network Segment** | VMs Event Viewer | No | Windows Security | Wireshark | Power Shell Script |
| **Micro-Segment** | VMs Event Viewer | No | Windows Security | NSX Traffic Capture | Power Shell Script |

**Table 4.4.6 Collection of Incident Data**

### 4.4.7 Conclusive Comparison of Micro-Segmentation vs Network Segmentation

To measure factors; I am choosing sliding scale from 1-5 as per following benchmark.

| Parameter | Scale | |
|---|---|---|
| | Micro-Segment | Network Segment |
| **Granular Segments** | 5 | 1 |
| **Access Control** | 5 | 2 |
| **Dynamic Security Policy** | 5 | 0 |
| **Detection and Response** | 4 | 2 |
| **Scalability** | 5 | 2 |
| **Flexibility** | 5 | 1 |
| **Adaptability** | 5 | 1 |
| **Latency** | 4 | 2 |
| **Complexity** | 4 | 2 |
| **Throughput** | 2 | 4 |
| **Technical Expertise** | 5 | 3 |
| **Configuration and Deployment** | 5 | 2 |
| **Robust Auditing Capabilities** | 5 | 2 |
| **Deployment Cost** | 3 | 5 |
| **Operational Cost** | 2 | 5 |

**Table 4.4.7 Sliding Scale Micro-Segmentation vs Network Segmentation**

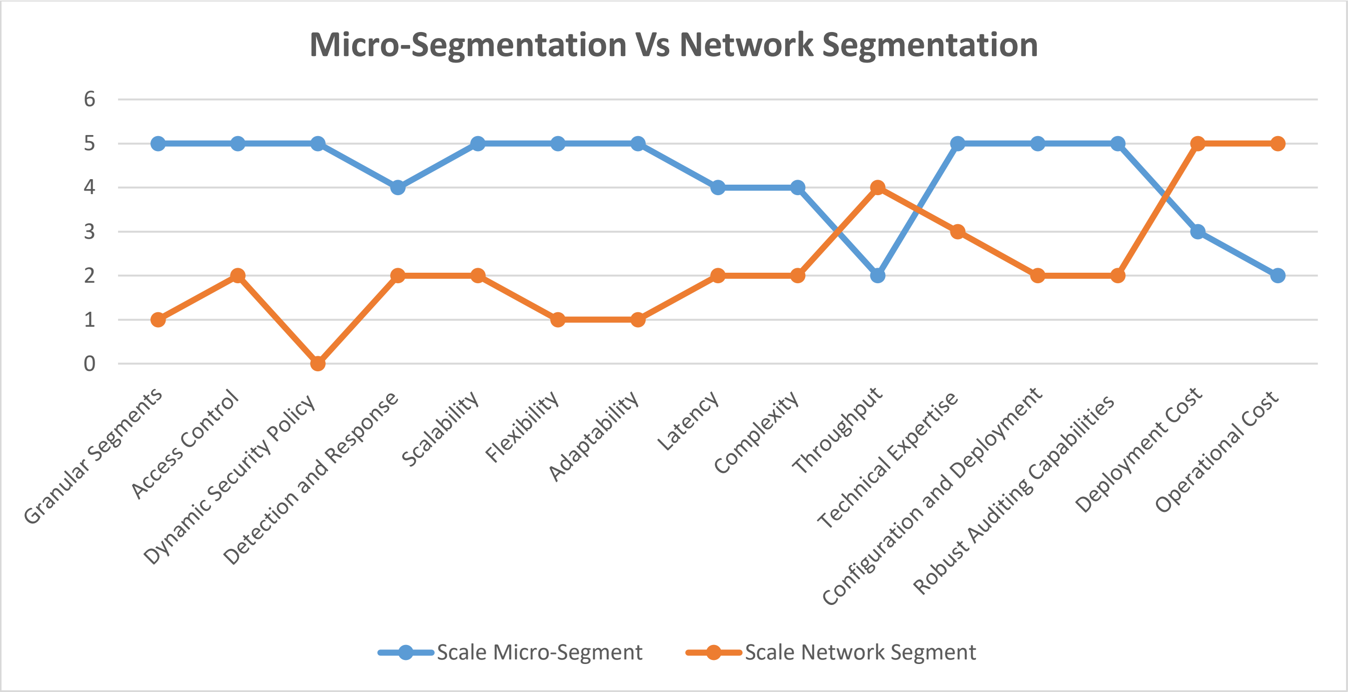


**Fig 4.4.2 Micro-Segmentation Vs Network Segmentation**

*Chapter 5*

# CONCLUSION

It is crucial for enterprises to put appropriate security measures in place as the threat landscape changes constantly. With its capacity to impose granular security controls and lower the threat surface of malware, micro-segmentation emerged as a potential strategy to take into consideration. Future study in this field might concentrate on improving the performance, scalability, and management elements of micro-segmentation as well as assessing how effective it is in various network contexts.

Overall, micro-segmentation shows potential as a potent weapon in the ongoing struggle against malware threats, offering improved security measures and boosting the resilience of network infrastructures.

## 5.1 Objectives Achieved

Following objective are achieved:

- Successfully assessed the combination of vmware NSX, a platform for software-defined networking, and Windows Defender, the built-in antivirus program for Windows operating systems. This connection made it easier to evaluate the security conditions of virtual machines and automatically issue security tags based on those evaluations.
- The research effectively assessed the efficiency of micro-segmentation in containing malware within individual segments and restricting lateral movement through thorough experimentation and analysis.
- The study achieved its goal of conducting an extensive comparison of network segmentation and micro-segmentation. To shed light on the benefits and drawbacks of each solution, a number of variables including security efficacy, scalability, complexity, performance impact, and auditing capabilities were assessed.
- In general, these goals show how successfully implementing and evaluating micro-segmentation as a method for lowering the threat surface of malware can help network administrators and security experts make wise decisions about securing their network environments.

## 5.2 Limitations

The limitations are as follows:

- Protecting the network and workloads within the segmented environment is the main goal of micro-segmentation. It does not offer complete defence against all threats, including intrusions coming from outside the segmented network or compromised endpoints there. For complete protection, a comprehensive security plan that incorporates micro-segmentation with other security measures is necessary.
- Malware may spread across segments or receive insufficient isolation if policies are not correctly set or if there are configuration errors. To make sure that policies adhere to security requirements, regular policy review and testing is essential.
- Micro-segmentation implementation and management can be difficult and time-consuming. It necessitates thorough preparation, configuration, and continual network traffic and security policy monitoring. A greater number of policies may need to be managed as a result of the segmentation's finer granularity, necessitating specialized resources and knowledge.

## 5.3 Future Direction

The future directions could be as follows:

- Micro-Segmentation should be combined with advanced threat detection and response systems. Look into the application of anomaly detection, behavioral analytics, and machine learning to improve detection and reaction in segmented contexts. **The security posture can be strengthened further by developing techniques for dynamically adapting security policies based on real-time threat intelligence.**
- Investigate ways to combine micro-segmentation with security information and event management (SIEM) technologies and threat intelligence feeds. Examine how threat information can be used in segmented environments to enable proactive defence, identify emerging threats, and dynamically alter security rules.
- Contextual SDP (Software defined perimeter) is new concept evolving, where network access to resources are dynamically defined for each user and its integration with micro segmentation.

## 5.4 Concluding Remarks

Micro-segmentation is a useful technology in the ongoing conflict with ever changing malware threats. Organizations can improve their network security posture and safeguard important assets in a constantly shifting threat environment by utilizing its advantages and resolving its limits. The success and durability of micro-segmentation as a vital part of contemporary cybersecurity methods will depend on ongoing innovation, research, and collaboration.

## References

[1] N. Sheikh, M. Pawar and V. Lawrence, "Zero trust using Network Micro Segmentation," *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2021, pp. 1-6, doi: 10.1109/INFOCOMWKSHPS51825.2021.9484645.

[2] William R. Simpson and Kevin E. Foltz, “Network Segmentation and Zero Trust Architectures” Proceedings of the World Congress on Engineering 2021 WCE 2021, July 7-9, 2021, London, U.K., ISBN: 978-988-14049-2-3

[3] Huang, Dijiang & Chowdhary, Ankur & Pisharody, Sandeep. (2018). Microsegmentation: From Theory to Practice. 10.1201/9781351210768-8.

[4] F. M. Alotaibi and V. G. Vassilakis, "SDN-Based Detection of Self-Propagating Ransomware: The Case of BadRabbit," in *IEEE Access*, vol. 9, pp. 28039-28058, 2021, doi: 10.1109/ACCESS.2021.3058897.

[5] Rouka, Elpida & Birkinshaw, Celyn & Vassilakis, Vassilios. (2020). SDN-based Malware Detection and Mitigation: The Case of ExPetr Ransomware. 10.1109/ICIoT48696.2020.9089514.

[6] Tank, Darshan & Aggarwal, Akshai & CHAUBEY, NIRBHAY. (2020). A Method for Malware Detection in Virtualization Environment. 10.1007/978-981-15-6648-6_21.

[7] Peter Friedrich Klemperer, PhD thesis Carnegie Mellon University Pittsburgh, PA, 2015.

[8] M. Mujib and R. F. Sari, "Performance Evaluation of Data Center Network with Network Micro-segmentation," 12th International Conference on Information Technology and Electrical Engineering (ICITEE), pp. 27-32, doi: 10.1109/ICITEE49829.2020.9271749, 2020.

[9] C. Gan, Q. Feng, X. Zhang, Z. Zhang and Q. Zhu, "Dynamical Propagation Model of Malware for Cloud Computing Security," in IEEE Access, vol. 8, pp. 20325-20333, 2020, doi: 10.1109/ACCESS.2020.2968916.

[10] Bays, L.R., Oliveira, R.R., Barcellos, M.P. et al. Virtual network security: threats, countermeasures, and challenges. J Internet Serv Appl 6, 1 (2015).

Tank, Darshan & Aggarwal, Akshai & CHAUBEY, NIRBHAY. (2020).

[11] Scheme Based on Zero Trust Architecture," ISCTT 2021; 6th International Conference on [11 ]Information Science, Computer Technology and Transportation, 2021, pp. 1-4.

[12] Klein, Dave (2019). Micro-segmentation: securing complex cloud environments. Network Security, 2019(3), 6–10. doi:10.1016/S1353-4858(19)30034-0

[13] Lior Rokach, Yuval EoVICI ACM computimng survey, Vol. 52, No. 5, Article 88. Publ;ication 2019.

[14] Leon, R.S., Kiperberg, M., Leon Zabag, A. et al. Hypervisor-assisted dynamic malware analysis. Cybersecur 4, 19 (2021). https://doi.org/10.1186/s42400-021-00083-9

[15] E. Rouka, C. Birkinshaw and V. G. Vassilakis, "SDN-based Malware Detection and Mitigation: The Case of ExPetr Ransomware," 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), 2020, pp. 150-155, doi: 0.1109/ICIoT48696.2020.9089514.

[16] Williams, J., Legg, P. Investigating Malware Propagation and Behaviour Using System and Network Pixel-Based Visualisation. SN COMPUT. SCI. 3, 53 (2022).

[17] Ensemble Learning at Hypervisor," 2018 IEEE Global Communications Conference (GLOBECOM), 2018, pp. 1-6, doi: 10.1109/GLOCOM.2018.8648070.

[18] https://resources.infosecinstitute.com/topic/how-malware-detects-virtualized-environment-and-its-countermeasures-an-overview/

[19] A Method for Malware Detection in Virtualization Environment. 10.1007/978-981-15-6648-6_21.

[20] NIST Publishes SP 800-215: Guide to a Secure Enterprise Network Landscape November 17, 2022.

[21] Arifeen, Murshedul & Petrovski, Andrei & Petrovski, Sergey. (2021). Automated Microsegmentation for Lateral Movement Prevention in Industrial Internet of Things (IIoT). 1-6. 10.1109/SIN54109.2021.9699232.

[22]

[23] Sneha, Sekaran & Lakshmanan, Malathi. (2015). A Survey on Malware Propagation Analysis and Prevention Model. International Journal of Advancements in Technology. 06. 10.4172/0976-4860.1000148.

[24] Tank, Darshan & Aggarwal, Akshai & CHAUBEY, NIRBHAY. (2020). A Method for Malware Detection in Virtualization Environment. 10.1007/978-981-15-6648-6_21.

[25] Zhang, Yu & Bhargava, Bharat. (2008). Fibonacci Modeling of Malware Propagation.

[26] Williams, Jacob & Legg, Phil. (2022). Investigating Malware Propagation and Behaviour Using System and Network Pixel-Based Visualisation. SN Computer Science. 3. 10.1007/s42979-021-00926-9.

[27] Ahmad, Muhammad & Woodhead, Steve & Gan, Diane. (2016). The V-Network Testbed for Malware Analysis. 10.1109/ICACCCT.2016.7831716.

https://attack.mitre.org/techniques/T1587/001/